國立臺灣大學電機資訊學院資訊工程學研究所

碩士論文

Graduate Institute of Communication Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

展頻隱藏性聲音通訊之二層強化及其應用

A Two-Level Enhancement for Spread Spectrum Based Hidden Aerial Acoustic Communications and its Applications

林志宏

Chih-Hung, LIN

指導教授：吳家麟 博士

Advisor: Ja-Ling Wu, Ph.D.

中華民國 99 年 7 月

July, 2010

誌謝

能完成這篇論文，我首先要特別感謝我的指導教授吳家麟老師，接著要感謝台大資訊所多媒體實驗室的每一位成員對我的支持與鼓勵。特別是同屬於吳老師門下，博士班的 Jerry、嘉祐學長、小白學姐、小卡學長、阿孜孜學姐，以及臨坐的萊曼學長；碩二的瀚平、傳甡、G姊、病毒、Boris及來自捷克的史安德；碩一的學弟妹們，謝謝大家。另外，還要特別感謝實驗室的助理謝哥，為我張羅器材，讓我無後顧之憂地研究。

林志宏　謹誌

中華民國九十九年七月

中文摘要

由於手持式運算設備的普及,隱藏性聲音通道是一個低成本的資訊傳遞工具。過去，此類通訊系統有兩派主流作法，其一利用正交分頻多工技術,傳輸效果好，但聲音品質差；其二利用展頻浮水印的技巧，將資訊隱藏在聲音當中，此法聲音品質較佳，但傳輸能力有限。

我們重新測量聲音通道的表現及特徵，進而針對展頻浮水印之通訊系統提出了兩個不同層次的強化方法：通訊方法階段採用了適應分頻技術，在不影響聲音品質的前提下，增加了傳輸能力；而在資料處理階段，利用霍夫曼編碼提昇傳輸效率。最終在整體表現上,我們維持了好的聲音品質,並達到了較好的傳輸速率。

關鍵字：展頻浮水印、正交分頻多工、通訊系統。

ABSTRACT

Due to the growing popularity of handheld devices, using sound to pass information from audio source to mobile becomes a possible application. Although the throughput offered by channel is relatively low and the quality of audio is not as good as the original, it provides a costless solution for communication with mobile devices through acoustic channel. Many interesting applications based on such communication systems have been proposed.

To reach the goal, spread spectrum watermarking technologies have been considered before, but the achieved throughput is very low. Another approach called “Acoustic OFDM” is proposed. The achieved transmission rate is superior to that of the SS-based method, but its associated quality of audio is controversial.

We proposed two-level enhancement for spread spectrum watermarking based communication over the hidden aerial acoustic channel. It makes the throughput more than “Acoustic OFDM” method, but keeps the quality of audio as good as SS-based method.

Keywords: acoustic channel, data hiding, Huffman codes, synchronization, OFDM, frequency division, communication system.

CONTENTS

LIST OF FIGUREURES

LIST OF TABLES

Chapter 1 Introduction

The interest of data transmission over hidden acoustic channel has sprouted in recent years. In this kind of schemes, the audio played by ordinary loudspeakers has been modified to embed data. Once the microphone receives the signal through aerial acoustic channel, the receiver can perform the adaptive process to extract the message which is hidden in the sound.

Due to the growing popularity of handheld devices, using sound to pass information from audio source to mobile becomes a possible application. Although the throughput offered by channel is relatively low and the quality of audio is not as good as the original, it provides a costless solution for communication with mobile devices through acoustic channel.

Many interesting applications based on such communication systems have been proposed. Two examples are exhibited in the following:

A. Transfer coupons or promotion information to client, as shown in Figure 1-1.

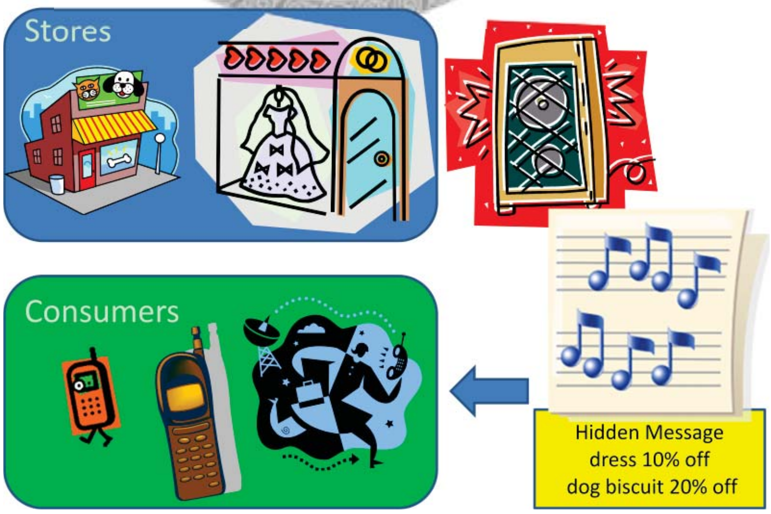


Figure 1-1. Application scenario I – (Promotions / coupons)

In a shopping mall, stores can pass promotion information to consumers through music played by loudspeakers. It is very possible to stimulate the purchasing behavior of consumer.

B. Transfer traffic information to car drivers. As shown in Figure 1-2.

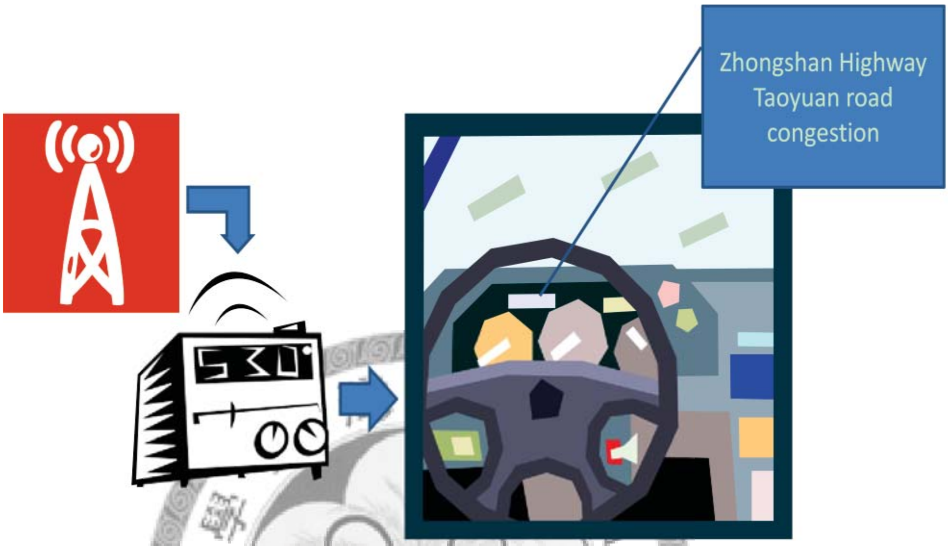


Figure 1-2. Application scenario II – (Traffic information)

There are many broadcasting stations reporting the instant traffic information to car drivers. The programs usually play music with a list, and the host announces traffic information during the break between two songs. At that time, driver needs to stop enjoying music and pays his attention to judge whether the traffic jam is on his way or not. Most of messages are not useful, possibly very far from the driver. The dilemma can be solved by the proposed transmission system, in which the information is passes through the audio (or the hidden acoustic channel). A receiver can extract and filter out useless information, and then display the useful information on dashboard.

Nevertheless, the communication over acoustic channel through ordinary loudspeaker to microphones inevitably suffers distortion channel by the environmental noise. Meanwhile, audio encounters D/A – A/D conversions which are mainly conducted on the sound cards. Although raising the strength of hidden signal could make the communication more robust; however, the interference to human hearing increases at the same time. Shortening the transmission distance may reduce the environmental interference, but very short distance is not applicable to most applications. Finally, the achievable data-rate is the most important consideration of the system. Low data rates take many seconds to transmit even a short message such as a single word. Judging from the above discussions, the requirements can be summarized as follows:

A. Robustness and Transmission Distance

Transmission should be as robust as possible, since most applications cannot tolerate frequent errors. The transmission distance is another impacting factor for transmission capability. In many situations, short transmission distance is inconvenient and impractical. This issue is affected by the volume of played sound and the strength of embedded signal.

B. Fidelity

In some applications, it is important to keep the embedded audio in good quality. For instance, in a shopping mall, a customer cannot tolerate music sounds with a lot of noise. To keep fidelity, the strength of the embedded signal should be constrained.

C. Throughput Rate

Data rate of the delivered information must be high enough for the selected application scenarios. For example, in the text transmission mode, users shall be able to read at the normal reading speed. If the total signal strength is constrained, increasing throughput means decreasing a transmission unit of the power. Enhancing throughput will reduce robustness of a system or cause the bad fidelity.

To reach the goal, many watermarking technologies have been considered before, such as echo hiding [7], phase coding [8] and spread spectrum [1][3] watermarking schemes. Unfortunately, neither echo hiding nor phase coding is robust for acoustic links. Although the spread spectrum technique can succeed in transmitting data, the achieved throughput is very low. In [3], given above 90% correct extraction ratio, the highest achievable throughput is only 213 bits per minute (3.55 bps). In [1], the error correction mechanisms have been taken into account, and DCT is adopted instead of DFT as the transformation of the data. The achievable of in [1] is 76.5625 bps with transmission distance less than two meters.

In [5], another approach called “Acoustic OFDM” is proposed. It applies OFDM technique to the high frequency portion of the original audio. The achieved transmission rate is about 240 bps with distance less than three meters, which is superior to that of the SS-based (Spread Spectrum Watermarking based) method. This approach provides good performance in transmission, but its associated quality of audio is controversial.

Both SS-based method and Acoustic OFDM are state-of-the-art techniques for aerial acoustic communication, because each of them has their own pros and cons. SS-based method wins in fidelity, and Acoustic OFDM wins in throughput. Figure 1-3 briefly shows the behavior characteristics of the two methods.

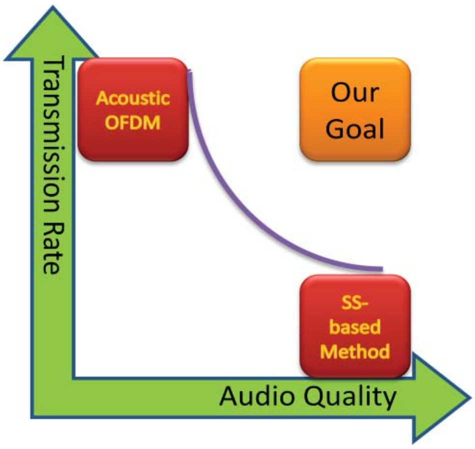


Figure 1-3. The characteristics of two state-of-the-art aerial acoustic communication methods

To transmit more data through the system under good audio quality, it must increase the throughput of SS-based method or enhance the fidelity of Acoustic OFDM approach. Because the required band-cut process in Acoustic OFDM abandons high frequency signals, it is not easy to enhance the fidelity. In order to increase the throughput, we propose two enhancing approaches of SS-based method in different levels.

According to the channel response described in chapter 3, we redesign the embedding procedure to make the channel more efficient. In the other words, it can transmit more bits at the same time through the modulation-level enhancement. Because most applications of the system are in text-based transmissions, it is possible to save the bit-rate through effective coding approaches. For example, variable length coding (VLC) is a useful tool for increasing the information rate. Since the code length is not constant, the error propagation problem will happen once receiving an incorrect bit. The error propagation controlling mechanism is, therefore, introduced in the data-level enhancement.

In our experiments, under the same transmission distance, the information rate and audio quality of the proposed approach are better than that of Acoustic OFDM. In the proposed system, to transmit data at the rate of 302.25 bps, the largest transmission distance is about two meters; to transmit data less than three meters, the throughput is around 230 bps. Our system has also integrated with Huffman coding, which enhances the transmission rate about 50%.

The remaining parts of this thesis are organized as follows. Chapter 2 illustrates the general architecture of aerial acoustic communications and reviews related works including SS-based and Acoustic OFDM methods. Chapter 3 depicts the channel characteristics and responses of various devices. Chapter 4 discusses modulation-level enhancement. Chapter 5 exhibits the integration with Huffman encoding. Comparisons of the proposed expansion and the Acoustic OFDM are presented in Chapter 6 via extensive experiments. Conclusions and future research direction are provided in Chapter 7.

Chapter 2 Related Work

2.1 Hidden Acoustic Communication System Overview

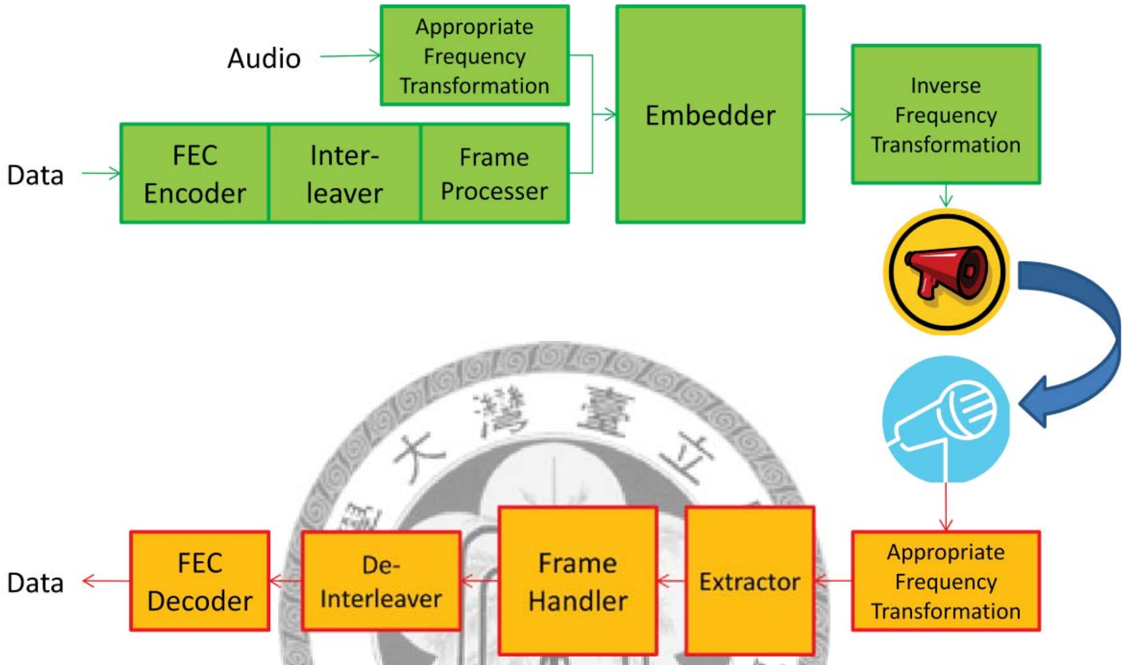


Figure 2-1. System Block Diagram

The overall process of a hidden acoustic channel based communication system is shown in Figure 2-1. It can be divided into 3 parts: audio preprocessing, data preprocessing and embedding/extracting.

A. Audio preprocessing

Data embedding methods usually are performed on the frequency domain, so the audio data should be transferred with the appropriate algorithm in a specific length first. In some methods, a masking threshold is used to constrain the modification of original audio. The masker is usually constructed according to a psychoacoustic model. Band pass filter is another necessary component here. It separates the audio signal into low frequency and high frequency parts.

B. Data preprocessing

In order to resist interferences caused by the channel (including D/A-A/D conversion and the acoustic noises), error controlling mechanisms are performed. Usually, error controlling mechanisms include: FEC (forward error correction) coding which adds redundant data to its messages and interleaving which is used to resist burst errors. For integrating every part of the system, some specific data structure must be considered. Some processes for data synchronization and the purposes are handled in this portion, for example, the specific data frame construction.

C. Embedding and extracting part

The method for embedding and extracting is the kernel of the system. On the sender side, the selected embedding method should be able to hide data into audio with a little modification. On the receiver side, the corresponding extracting method should be capable of extracting data from the receiving acoustic signals. The whole process includes waveform-level synchronization and symbol extraction. Throughput, robustness, and fidelity are the main guidelines considered in the overall system design.

2.2 Related Works

2.2.1 Spread spectrum based method

Spread spectrum watermarks are well-known for their good robustness and fidelity, but low information capacity, due to host interference, is the associated shortage. So far, the state-of-the-art for this kind of methods was proposed in [1]. It selects DCT-domain to embed data, and integrates error-controlling mechanisms, including Reed-Solomon codes and diagonal data interleaving, into the system.

A. Audio preprocessing

The input audio is separated into several windows with *N* samples. For each window, DCT is performed to get the frequency domain samples. The result of a process is represented by *N* coefficients with respect to DCT frequencies. The lower indexed coefficients stand for the lower frequency components, and the higher ones stand for the higher frequency signals.

The inaudibility criteria are gathered to limit the modification strength for every critical band. For each critical band, the maximum value is constrained according to the masking threshold of the band. A scaling factor is used to scale the masking threshold of each frequency band. The details of this method can be found in [3].

B. Data preprocessing

The input data are separated into *n*-bit symbols. These symbols are encoded by Reed-Solomon codes. After FEC encoding, the diagonal data interleaving is performed on the output symbols. The frame headers for providing frame-level synchronization are added. See Figure 2-2 which is the same as Figure 5 in [1].

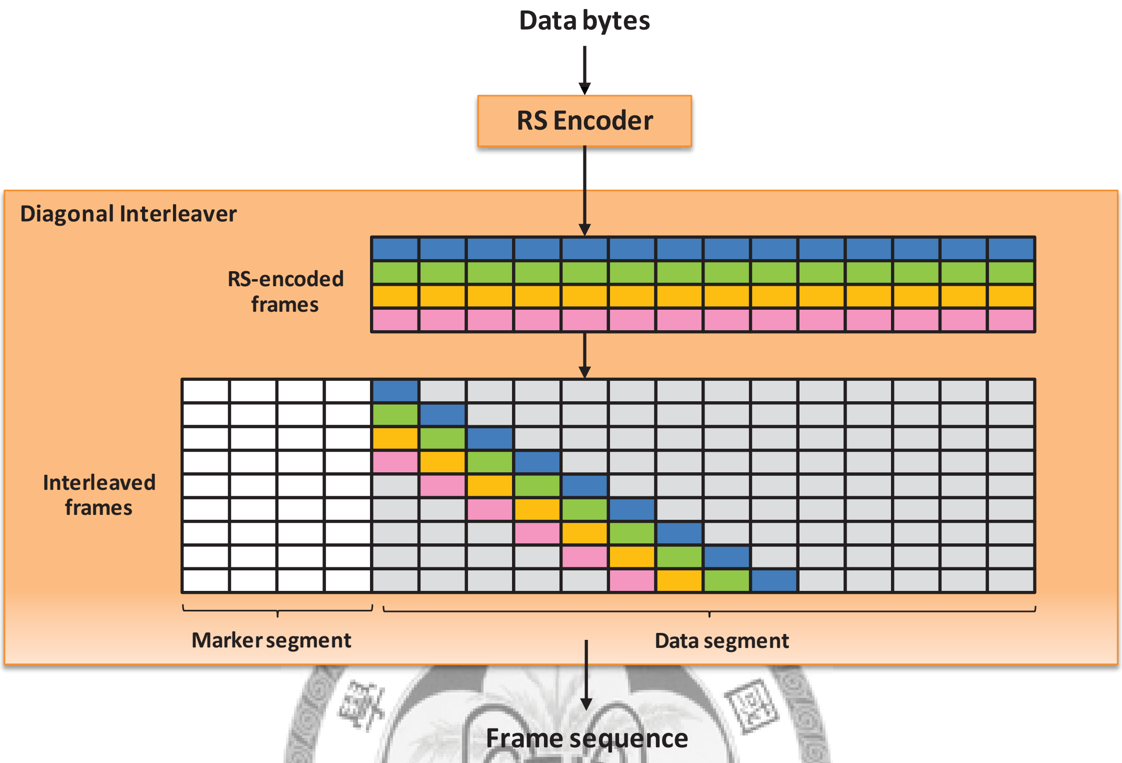


Figure 2-2. The diagonal data interleaver [1].

C. Embedding and extracting

Figure 2-3 illustrates the embedding procedure. The frame symbols are translated into codes from a designate codebook. The codebook contains $2^n + 1$ pseudorandom sequences, for n-bit symbols and the escape symbol. After getting the corresponding codes, the embedding strength is determined according to the prescribed masking threshold. Finally, the signals are embedded into the original audio in the frequency domain. The embedding procedure can be described as:

$$Y[k] = X[k] + A[k] \cdot C_j[k],\ \ k = \Delta \cdots, \Delta + l, \qquad (2.1)$$

where X denotes the original signal in the frequency domain, Y denotes the output signal, A denotes the masking threshold, C_j is the corresponding code, and k refers to the middle bands which are selected for embedding.

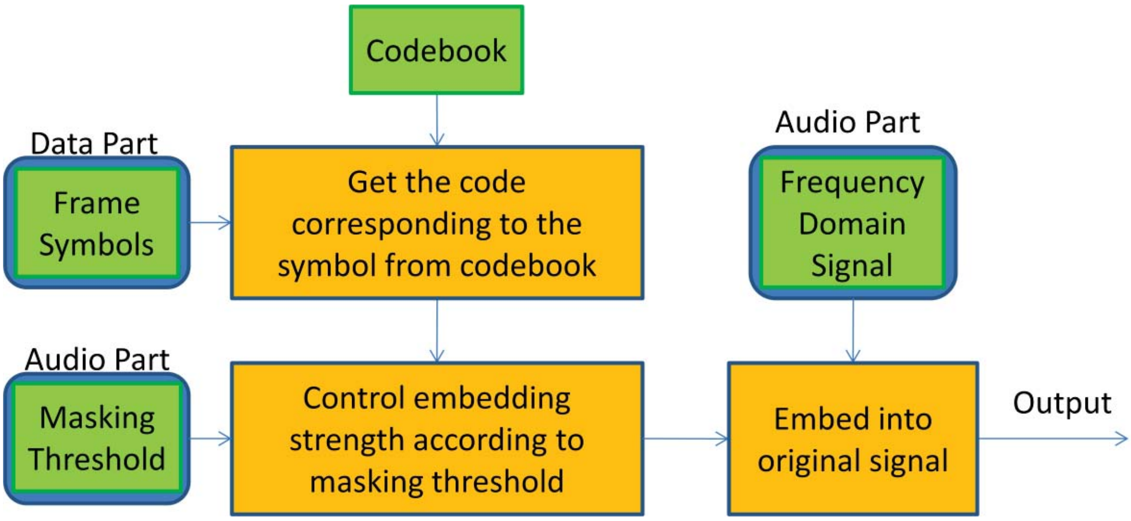

Codebook
Data Part
Frame Symbols
Get the code corresponding to the symbol from codebook
Audio Part
Frequency Domain Signal
Audio Part
Masking Threshold
Control embedding strength according to masking threshold
Embed into original signal
Output

Figure 2-3. The embedding procedure

The extracting procedure is followed by two levels of synchronization including: waveform-level synchronization and frame-level synchronization. In the waveform-level synchronization, the running DCT is performed on the receiving signals. For each run, the correlations between signals and codes in the codebook are calculated. The offset with the maximal correlation is regarded as the synchronized position, and the signals are extracted and transferred into symbols from there. After the waveform-level synchronization, there are many symbols in the buffer. The frame-level synchronization is operated to find the frame header which is treated as the starting point of a frame. While gathering enough frames, the inverse processes of data interleave and Reed-Solomon decoding are executed on the extracted symbols to extract the embedded data.

2.2.2 Acoustic OFDM [6]

This method uses OFDM technique to hide data in the higher frequency bands of the original audio. Unlike spread spectrum watermarking which spreads interferences into wide range of bands with low embedding strength, it embeds data through phase modification of selected carriers. The system is also integrated with convolution codes and data interleaving processes.

A. Audio preprocessing

The input audio is separated into several windows with N samples. For each window, FFT is performed to get the frequency domain samples. The result of a process is represented by N / 2 significant coefficients. The lower indexed coefficients stand for the lower frequency components, while the higher ones stand for the higher frequency signals.

A band-pass filter is used to separate the signal into low band and high band portions. For the high band signals, only the amplitude information needs to be kept which is regarded as the spectrum envelope for the carriers. For the low band signals, the masking threshold is needed for controlling the strength of the synchronization symbol.

B. Data preprocessing

For input bits, convolution codes are used to resist errors. In order to recover burst errors, the data are interleaved. A single data frame has N bits consisting of N_{symbol} OFDM symbols. In each symbol, there are N_{carrier} carriers. Data frames are transmitted in a repeated fashion. The last OFDM symbol is followed by the first OFDM symbol. The receiver can recognize boundaries of frames by checking the receipt of the same OFDM symbol.

C. Embedding and extracting part

In the lower band, it hides a synchronization symbol using spread spectrum watermarking techniques. In the high band, the selected orthogonal carriers are prepared for transmitting data. The Differential Binary Phase Shift Keying (DBPSK) is chosen to modulate the carriers. If the incoming bit is different from the past one, the phase of the carrier is shifted 180 degrees. For each carrier, the strength is controlled by the original high band amplitude information which is kept in the audio preprocessing. Figure 2-4, which is the same as Figure 2 in [6], shows the encoding procedure of Acoustic OFDM.

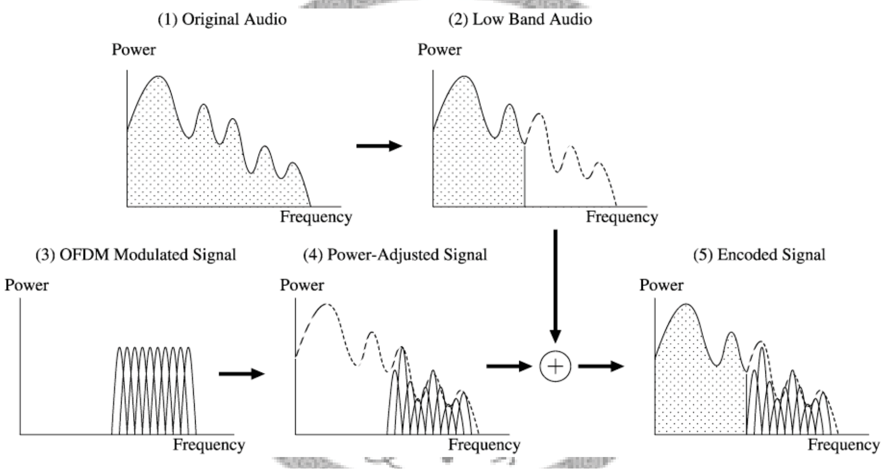


Figure 2-4. Acoustic OFDM encoding [6]

To eliminate the inter-symbol interference under multi-path conditions, Guard Interval (GI) is added. The GI, a copy of the last part of the data, preserves the orthogonality of carriers and the independence of subsequent OFDM symbols. Due to DBPSK, the phase is not continuous at the boundaries of ODFM symbols. The side lobe leakage caused by the phase discontinuity is reduced by applying a trapezoidal windowing. Figure 2-5, which is the same as Figure 3 in [6], shows the structure of Acoustic OFDM symbols.

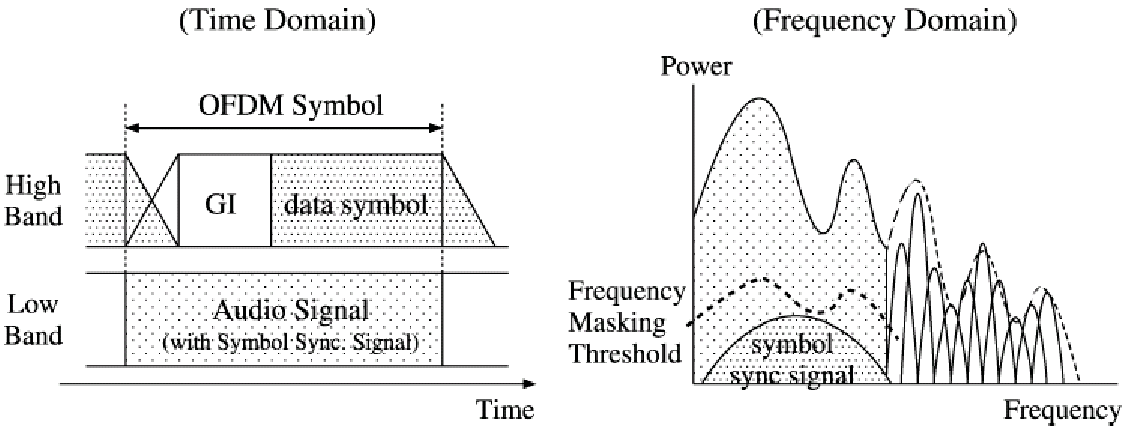


Figure 2.5. Structures of Acoustic OFDM symbols

In the extracting procedure, the receiver recognizes the synchronized point by detecting the synchronization symbol in the low band. The frame level synchronization is conducted by finding the first and the last symbols, which are the same.

Chapter 3 Channel Investigation

At the beginning of this work, we implement Acoustic OFDM with the configuration proposed in [6]. Unfortunately, after the system finished, only a few carriers in lower band can transmit data through the overall channel correctly. The high band signals suffer from severe distortions. The channel response may help us to adjust parameters of the system. So, we start to investigate the response of the aerial acoustic channel.

3.1 Channel Overview

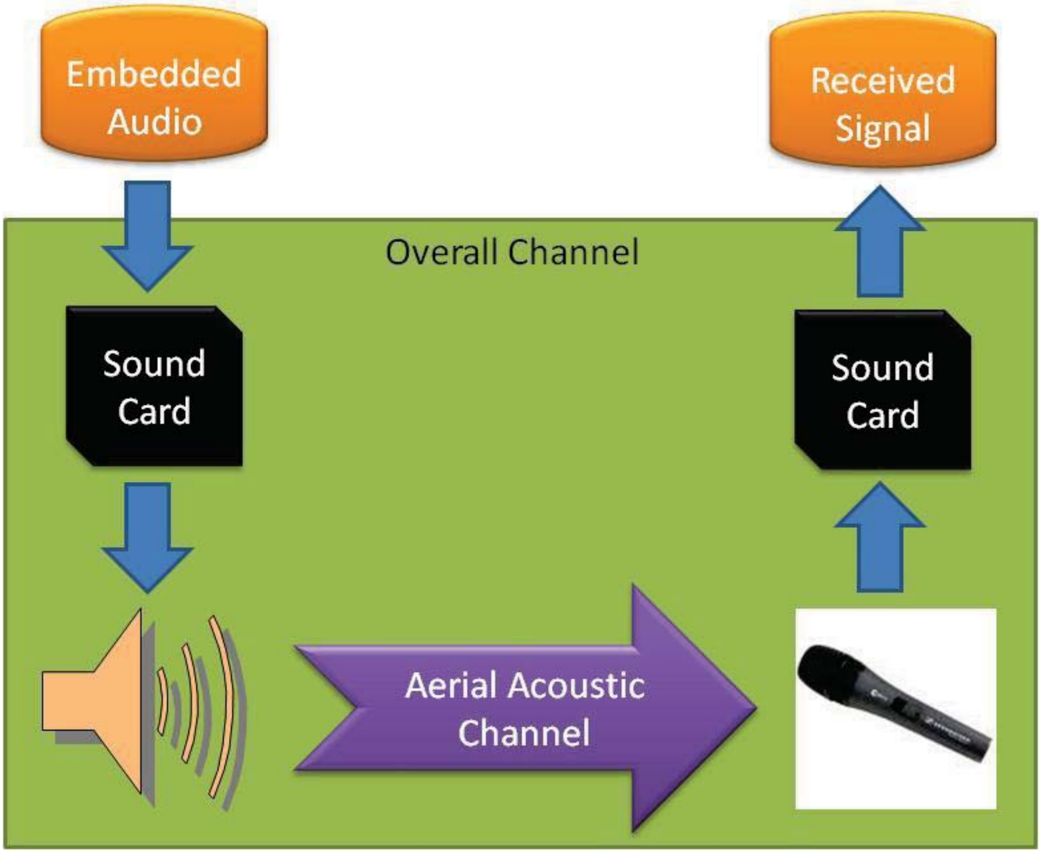


Figure 3-1. The aerial acoustic channel

On the sender side, data are embedded into the source audio, and then the embedded digital signal is sent to the sound card. After digital-to-analog conversion, the loudspeaker receives analog signal from the sound card. On the receiver side, the microphone receives sound and transfers it into analog signal. Analog-to-digital conversion is performed on the sound card. Finally, the receiving device extracts messages in the digital signal coming from the sound card.

Besides aerial acoustic channel, devices play important roles in determining the channel response. From Figure 3-1 and above discussion, the channel response of the aerial acoustic channel will be affected by the characteristics of the following three devices: Loudspeaker, Microphone, and Sound Card.

3.2 Channel Responses

In the beginning, we used cheap devices which are listed in Table 3-1. The overall channel response of this configuration is shown in Figure 3-2. Obviously, the response drops down at about 7200 Hz, and there is not any response over 8000Hz. This explains why the implementation of Acoustic OFDM was failed.

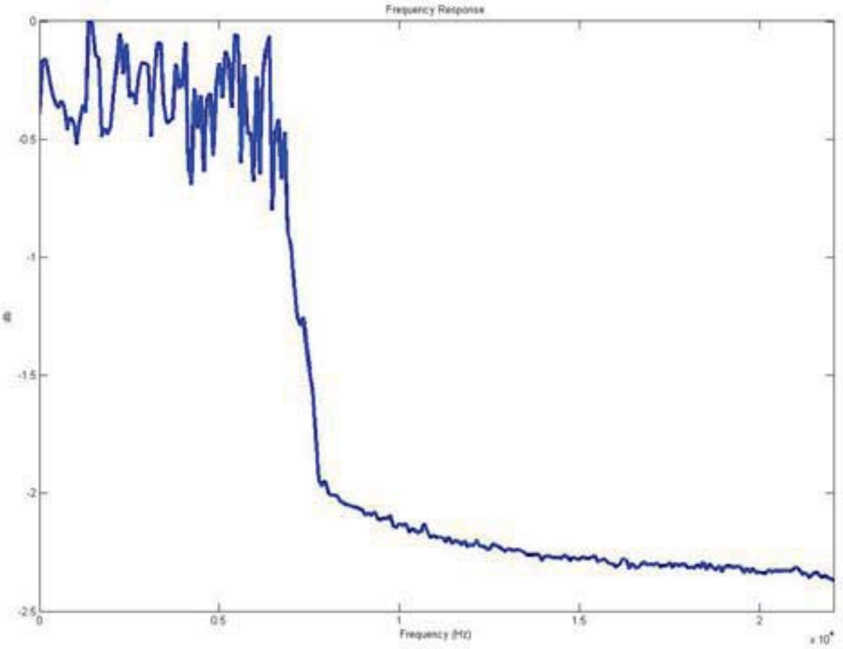


Figure 3-2. The channel response of configuration I listed in Table 3-1

Loudspeaker	KINYO PS-205 2.0 Multimedia Speaker
Microphone	Philips SHM1000/97
DA/AD	in Lenovo ThinkPad X61

Table 3-1. Our channel hardware configuration I

In [4], Nakashima et al. used better hardware devices in their experiment. In their report, the system with their devices has flat response in the frequency range 6000Hz ~ 10000Hz. Table 3-2 shows their configuration in details.

Loudspeaker	Fujitsu-Ten Eclipse TD508
Microphone	Sony ECM-360
DA/AD	in Dell Inspiron 5150

Table 3-2. Experimental hardware configurations used in [4].

For fair comparison, we prepared almost the same equipments, except the DA/AD device. Unfortunately, the channel response of this combination performs only a little better than that of the cheap devices. It still falls down at about 7200 Hz. So, we think the remaining problem is the DA/AD conversion. The frequency response of ThinkPad X61 built-in DA/AD conversion is shown in Figure 3-3. The dropping of the frequency response is mainly caused by the AD chip on the sound card, because people usually only need to record their low frequency voice in most situations.

In order to solve the problem, an external sound card, ASUS Xonar U1, is chosen to replace the DA/AD part of the channel. Its frequency response is shown in Figure 3-4 and the overall channel response is shown in Figure 3-5, in which the hardware configuration is given in Table 3-3. Notice that, from Figure 3-5, the flatness of the frequency band from 6000 Hz to 8000 Hz is much better than that in Figure 3-2. In other words, a much fair experimental environment is built for conducting OFDM based audio data embedding.

Loudspeaker	Fujitsu-Ten Eclipse TD508 II
Microphone	Sony ECM-360
DA/AD	ASUS Xonar U1

Table 3-3. Our channel hardware configuration II

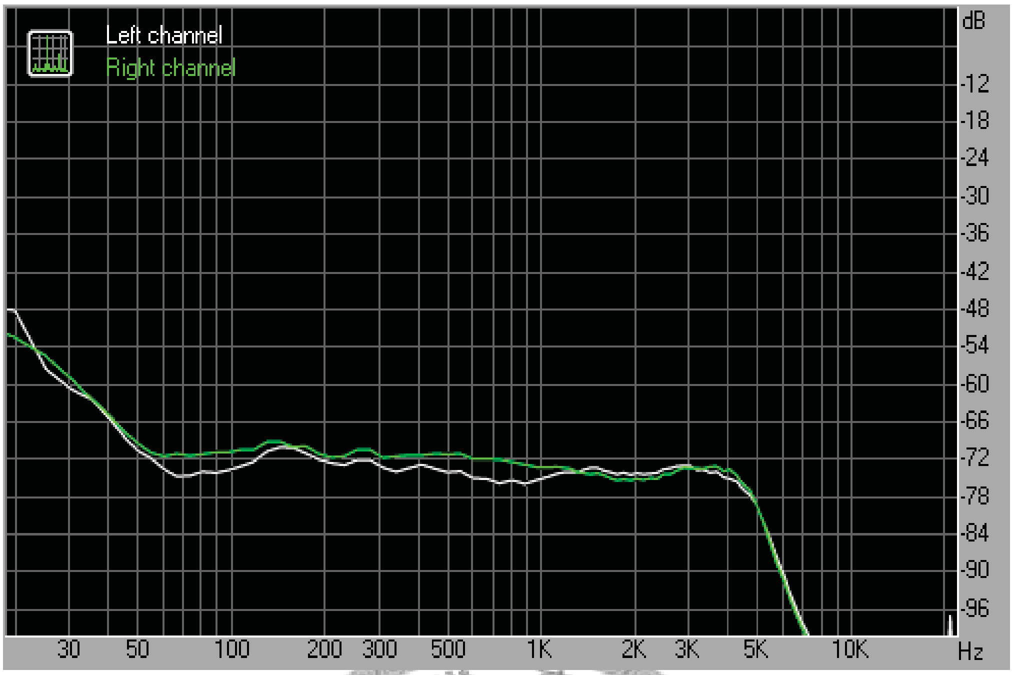


Figure 3-3. Frequency response of ThinkPad X61 built-in

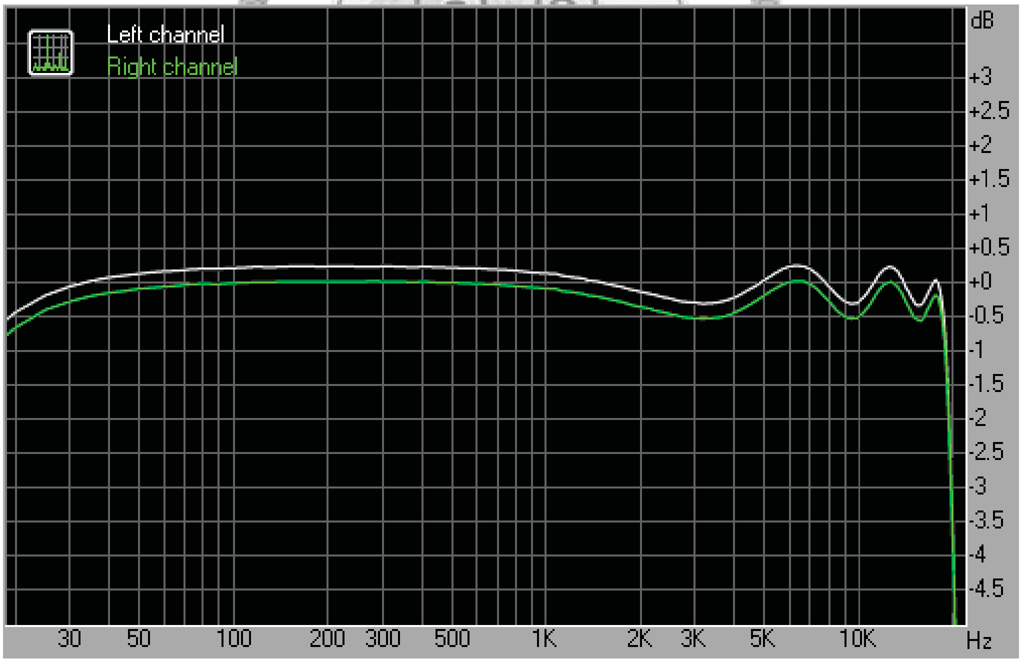


Figure 3-4. Frequency response of ASUS Xonar U1

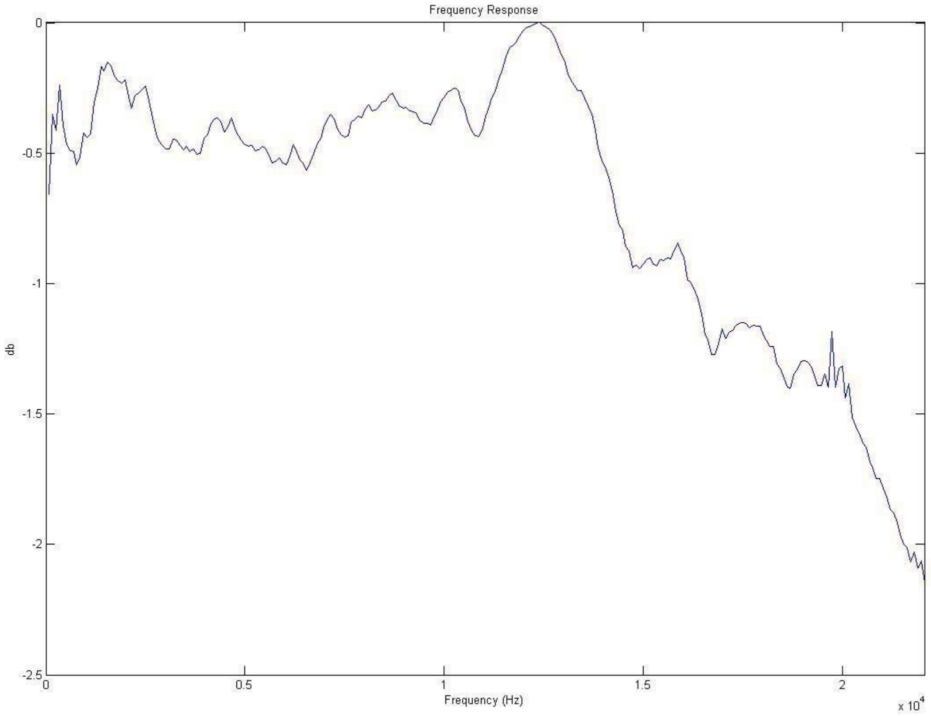


Figure 3-5. The channel response of configuration II listed in Table 3-3

Chapter 4 Modulation Level Enhancement

The channel responses have been investigated and shown in chapter 3. Obviously, with the hardware configuration I, Acoustic OFDM is not workable because it needs to transmit data over the frequency range 6400 ~ 8000 Hz. It is amazing that the method based on spread spectrum watermarking is still workable, when the configuration I is used. In [1], the data is hidden in the frequency range 5512.5 ~ 11025 Hz within 512 samples. In other words, the method can transmit data through the band of 5512.5 ~ 7200 Hz correctly, even if the configuration I is used. This result demonstrates a possibility to raise throughput of the method based on spread spectrum watermarking.

Increasing throughput means transmitting more bits in a time slot, so the simple idea is to embed more bits in a symbol which is represented by *N* samples. This idea is investigated in the following sections. A novel adaptive frequency division method is also proposed to deliver more bits per source symbol.

4.1 Increasing Codebook Size

In SS-based method, to transmit a symbol which contains *n*-bits in a window, the corresponding codebook should consist of 1 + 2^n elements. All elements in the codebook must have low correlation to each other. The receiver picks up the most correlated PN (pseudorandom number) sequence from the codebook, and extracts the corresponding embedded bits.

An example of symbol decoding procedure is shown in Figure 4-1, where *N* = 512, *n* = 4. The symbols labeled by a, b and c are the most correlated symbols to PN1, PN15, and PN2, respectively. The corresponding bits are “0010”, “1111”, and “0011”, so the receiving bits represented by the three symbols are “001011110011”.

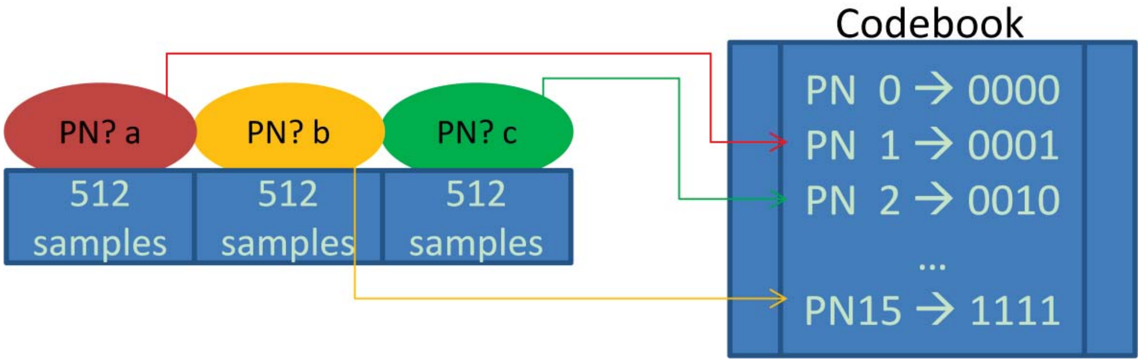


Figure 4-1. Symbol decoding procedures

Increasing bits per symbol can raise the throughput, but the codebook size is increasing exponentially at the same time. The more embedded bits in a symbol are ,the larger size of the codebook is. Meanwhile, the orthogonality between sequences in a codebook is decreasing, and it introduces a lot of computations on the receiver side. If one more bit is embedded in a symbol, the size of the codebook is doubled, so it grows exponentially. To reach doubled throughput, $2n$ bits are designated for a symbol, and the codebook size is enlarged to $2^{2n} + 1$. To make the bit-rate tripled, the codebook size will be enlarged to $2^{3n} + 1$. For triple transmission rate, the complexity of correlation computation is 2^{2n} times to that of the original one.

The high complexity of correlation computation makes the receiver spend a lot of time on waveform-level synchronization and symbol extraction. The idea of direct increasing embedded bits in a symbol is not suitable for speeding up due to the blocking of the receiver.

4.2 Frequency Division

4.2.1 Overview

Another idea is well-known in communication communities, called frequency division. Because a symbol needs only a narrow band (say 5512.5 ~ 7200 Hz) to transmit data from sender to receiver, it may divide available band (5512.5 ~ 14000 Hz) to multiple narrow subbands. For every *N* samples, we still embed *n* bits as the original method in each part. By this way, the throughput can be increased according to the number of divisions.

Figure 4.2 illustrates the frequency division approach with 3 divided parts. On the upper side, band1 is preserved for audio quality, and Band 2 is used to embed data as in [1]. On the lower side, we preserved the lowest band the same as the original, and divided available band into three narrow parts, which are labeled by 1, 2, and 3.

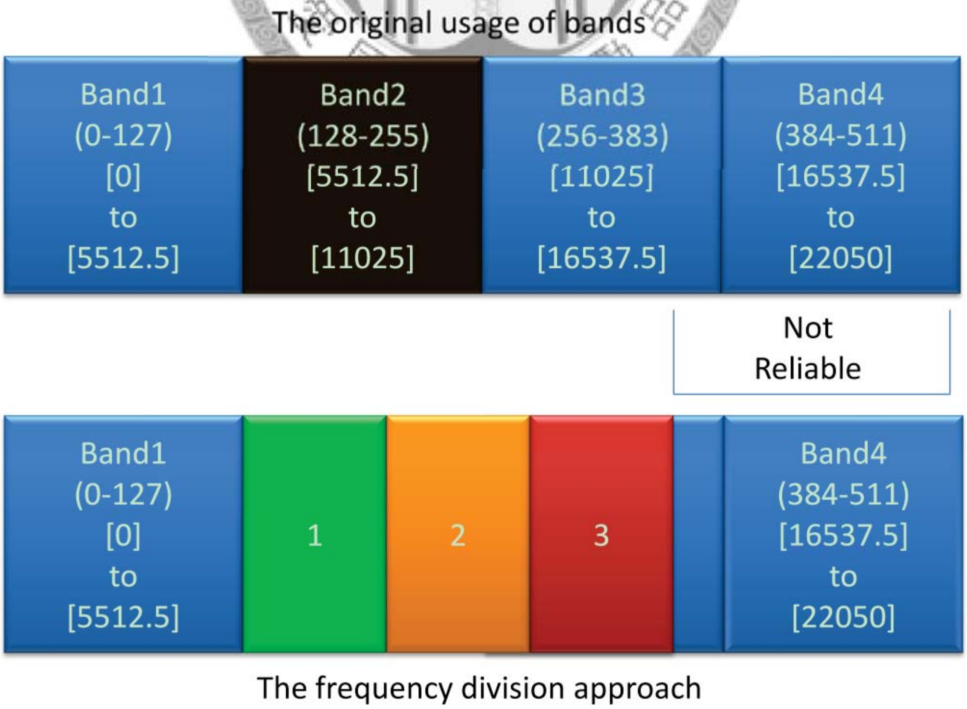

The original usage of bands
Band1 (0-127) [0] to [5512.5]
Band2 (128-255) [5512.5] to [11025]
Band3 (256-383) [11025] to [16537.5]
Band4 (384-511) [16537.5] to [22050]
Not Reliable
Band1 (0-127) [0] to [5512.5]
1
2
3
Band4 (384-511) [16537.5] to [22050]
The frequency division approach

Figure 4-2. The proposed frequency division approach with 3 divided subbands

To embed data to multiple divisions, three codebooks are used. In each codebook, the length of sequences is different from one another. The calculation complexity depends on the total embedding bandwidth, and it is the same as the original if the summations of transmitting bandwidths are equal. In this approach, the receiver only needs to extract data three times the original for receiving N samples.

4.2.2 Embedding Procedure

Let the coefficients of bands to be embedded are $B_{\mathrm{cut}} \ldots B_{\max}$. The coefficients $0 \ldots B_{\mathrm{cut}}$ are preserved for audio quality, and $B_{\max}$ is the highest embedding band coefficient which is selected according to the requirement of audio quality and/or channel capability. We divide the available band into dN divisions, sharing coefficients from bands B_{cut} to $B_{\max}$. For each division,

Let L_{avail} : length of available coefficients, that is $L_{\mathrm{avail}} = B_{\max} - B_{\mathrm{cut}}$, and

L_{s} (L_{avail}/dN) be the length of the available coefficients for a single division.

$B_{d1} = \{B_{\mathrm{cut}} \ldots (B_{\mathrm{cut}} + L_{\mathrm{s}})\}$, for the 1st division,

$B_{d2} = \{(B_{\mathrm{cut}} + L_{\mathrm{s}}) \ldots (B_{\mathrm{cut}} + 2 \times L_{\mathrm{s}})\}$, for the 2nd division,

…

$B_{dN} = \{(B_{\mathrm{cut}} + (dN - 1) \times L_{\mathrm{s}}) \ldots (B_{\mathrm{cut}} + dN \times L_{\mathrm{s}})\}$, for the dN^{th} division.

The corresponding PN sequences to be embedded in the current window are denoted as P_{d1}, P_{d2} …, P_{dN}. It starts to add data from B_{cut} with consecutive L_{s} coefficients for the 1st division, and the following L_{s} coefficients are used for the 2nd division, and so on. For embedded signals Y, the embedding procedure can be described as:

$$Y[k] = X[k], \qquad k < B_{cut} \qquad (4.1)$$

$$Y[k] = X[k] + A[k] \cdot P_{d1}[k], \qquad k \in B_{d1} \qquad (4.2)$$

$$Y[k] = X[k] + A[k] \cdot P_{d2}[k], \qquad k \in B_{d2} \qquad (4.3)$$

$$\ldots$$

$$Y[k] = X[k] + A[k] \cdot P_{dN}[k], \qquad k \in B_{dN} \qquad (4.4)$$

Figure 4-3 gives an example with *dN* = 3.

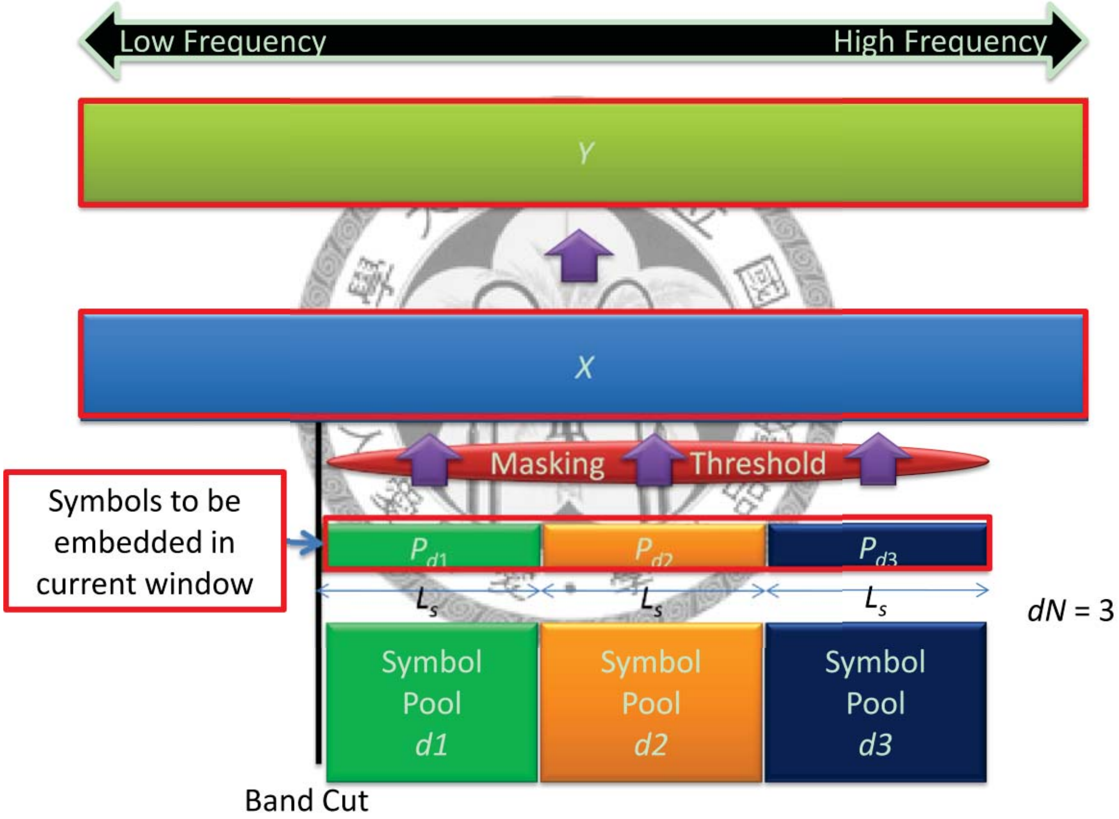


Figure 4-3. The embedding procedure for frequency division embedding, with *dN* = 3

4.2.3 Waveform-level synchronization

In applications of watermarking, synchronization attacks are a serious problem to most of the existing schemes. Once out-of-synchronization happens, the extractor will not know the starting point to decode the upcoming waveforms. Additionally, the different sampling of the sender and the receiver leads to inconsistence in time scales. Therefore, a periodic synchronization process is needed to alleviate this problem.

Figure 4-4 illustrates the waveform-level synchronization given in [1]. In that case, the all available bands are used to transmit one symbol. For a given offset, the correlation is calculated between the signals in the selected windows and the codewords within codebook. The summation of all correlations is regarded as the synchronization score of the offset. Trying all possible offsets, the one with the largest score is treated as the synchronized offset.

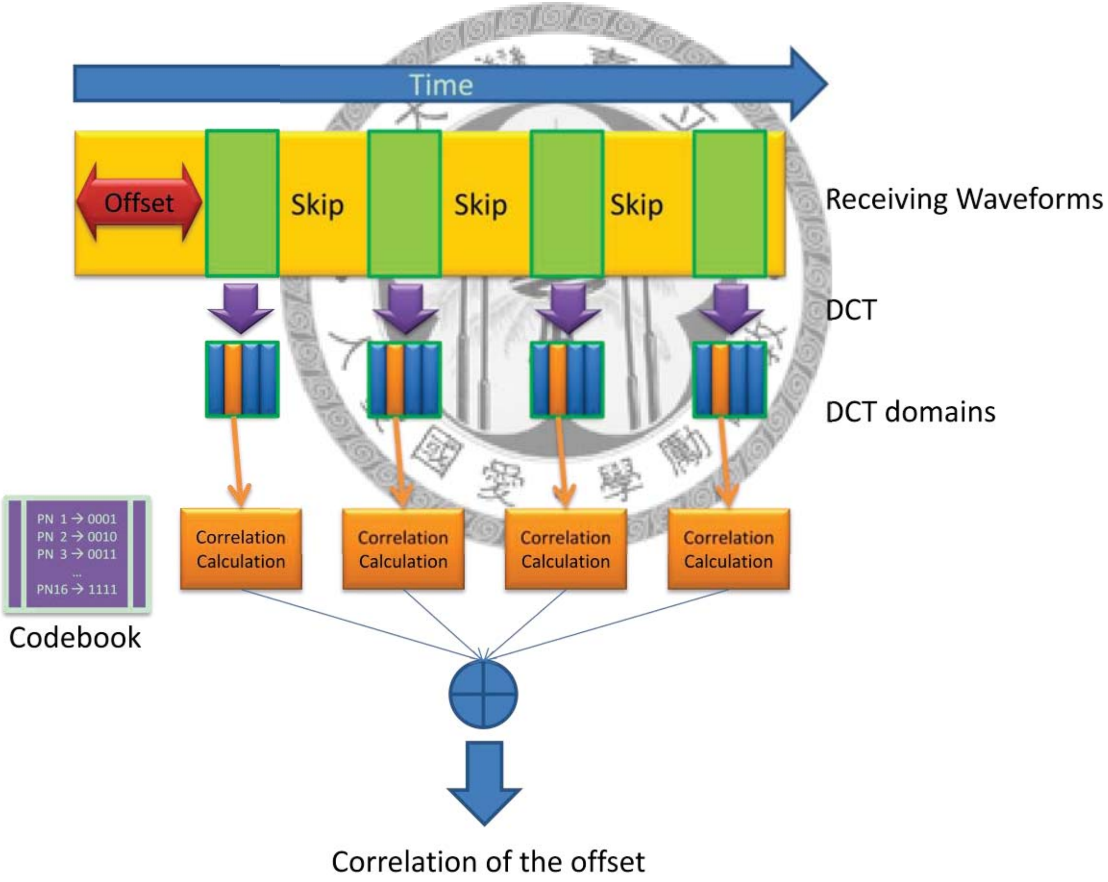


Figure 4-4. Waveform-level synchronization in [1]

To synchronize multiple divided parts, the correlation within a window is the summation of correlations between each division and the corresponding codebook. For an offset, the correlation is now the summation of all correlations of selected windows. The offset with the largest synchronization score is selected as the synchronized offset. See Figure 4-5 for the details.

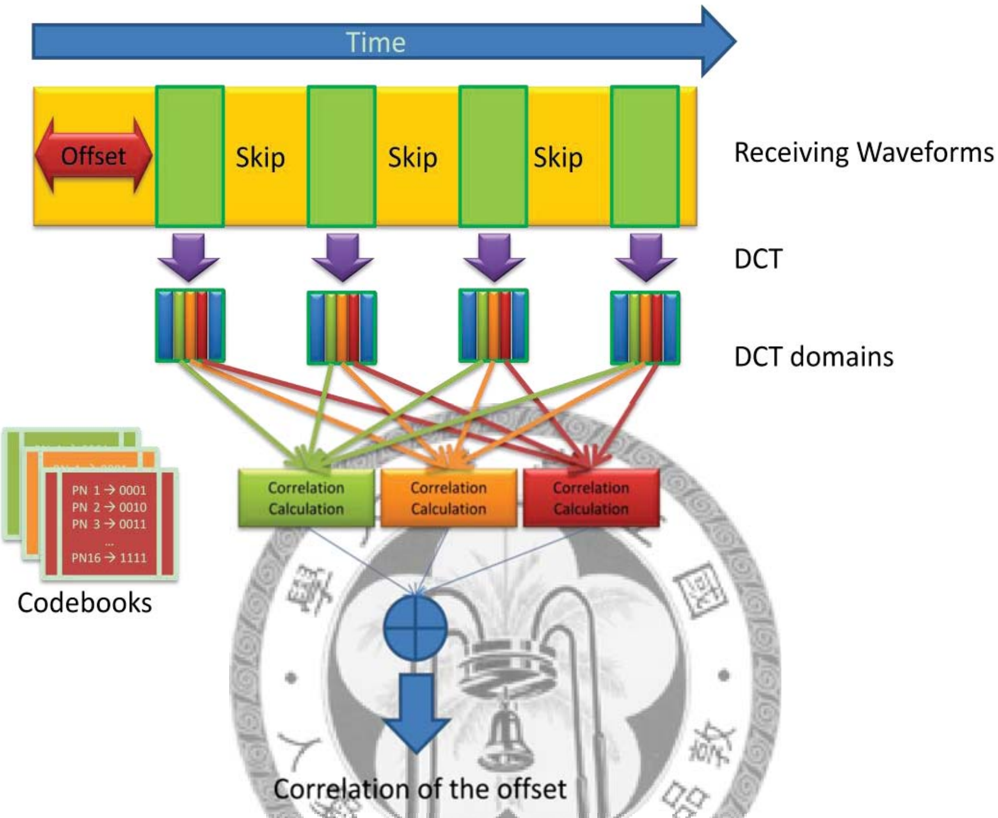


Figure 4-5. Waveform-level synchronization for multiple divided parts

4.2.4 Extraction procedure

For the synchronized offset, the data hidden in the consecutive windows of the signal will be extracted. The correlations between each window and PN sequences are calculated, and the sequence with the highest correlation score is regarded as the hidden symbol. The correlation calculation is similar to that of [1], but multiple symbols are extracted from the same window now.

4.2.5 Disadvantage

Unfortunately, channel response is not always flat in each subband. Besides environment noise, different hosts make different interferences to the embedded signals. The selective fading (as shown in Figure 4-5) makes error happen more frequent in the higher bands. Although the lower band transmission is robust, the higher band errors cause higher error rate for the overall system.

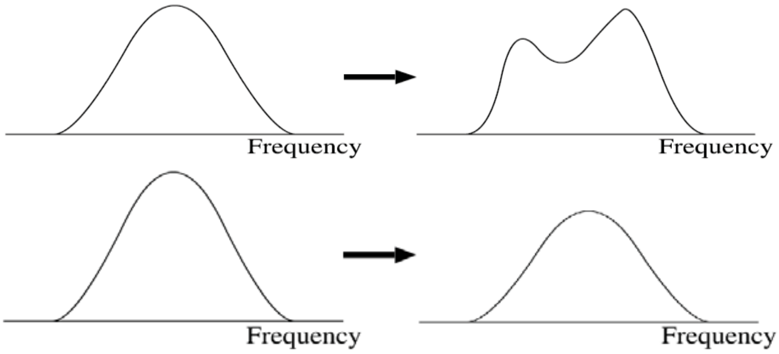


Figure 4-6. Flat fading (upper) and selective fading (lower)

4.3 Adaptive Frequency Division

4.3.1 Overview

To solve the problem caused by selective fading, the adaptive frequency division is proposed. The basic idea is to share robustness with every divided part. It is somewhat like frequency hopping techniques and it separate continuous frequency band into multiple orthogonal carriers in a repeated fashion. An illustrative implementation of the proposed adaptive frequency division, with 3 divided parts, is shown in Figure 4-7.

In this fashion, the fading of a division is similar to one another. Although the robustness of some division reduces slightly, the overall system becomes more robust.

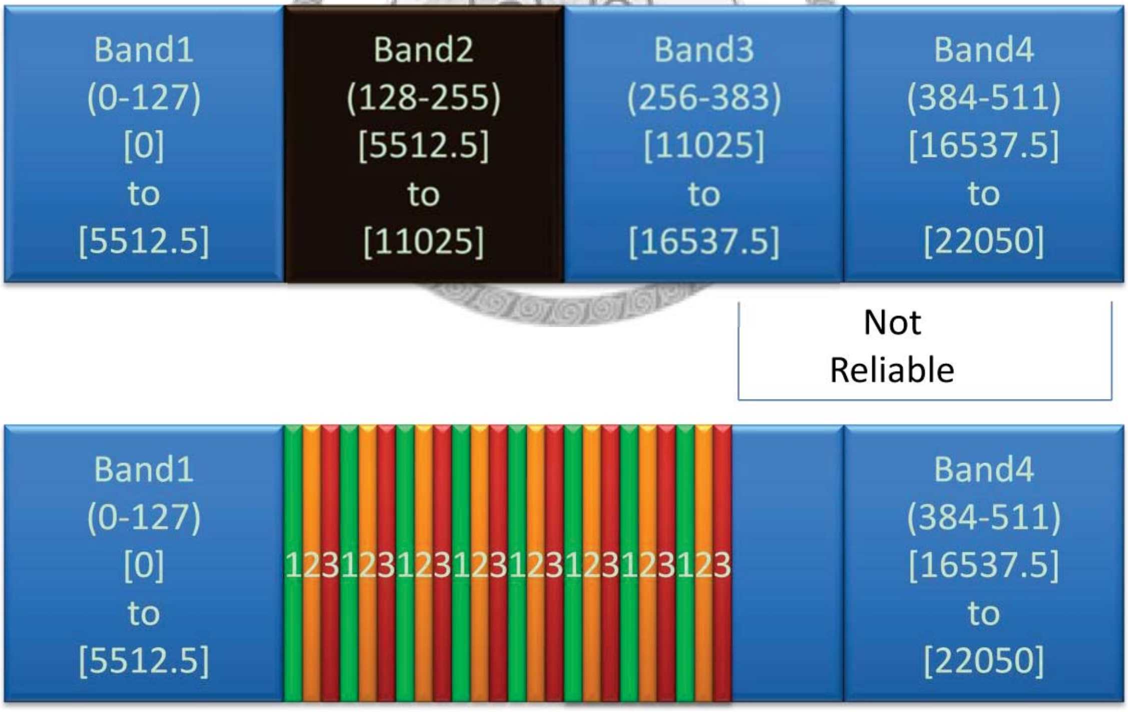


Figure 4-7. The adaptive frequency division approach with 3 divided parts

4.3.2 Embedding procedure

Let the coefficients of the bands to be embedded are B_{cut} … B_{max}, and we divide it into dN divisions. For each division, data are embedded into interlaced coefficients which are described as follows.

Let L_{avail} be the length of available coefficients, that is $L_{\text{avail}} = B_{\text{max}} - B_{\text{cut}}$, and

L_{s} (L_{avail}/dN) : length of the available coefficients for a single division.

$B_{d1} = \{B_{\text{cut}} + dN \times k \mid k = 0 \ldots L_{\text{s}} - 1\}$, for the 1st division,

$B_{d2} = \{B_{\text{cut}} + dN \times k + 1 \mid k = 0 \ldots L_{\text{s}} - 1\}$, for the 2nd division,

…

$B_{dN} = \{B_{\text{cut}} + dN \times k + (dN - 1) \mid k = 0 \ldots L_{\text{s}} - 1\}$, for the dNth division.

The corresponding PN sequences to be embedded in the current window are denoted as P_{d1}, P_{d2} …, P_{dN}. For the embedded signal Y, the embedding procedure can be described as:

$$Y[k] = X[k], \qquad k < B_{cut} \qquad (4.5)$$

$$Y[k] = X[k] + A[k] \cdot P_{d1}[k], \qquad k \in B_{d1} \qquad (4.6)$$

$$Y[k] = X[k] + A[k] \cdot P_{d2}[k], \qquad k \in B_{d2} \qquad (4.7)$$

…

$$Y[k] = X[k] + A[k] \cdot P_{dN}[k], \qquad k \in B_{dN} \qquad (4.8)$$

Figure 4-8 presents an example with $dN = 3$.

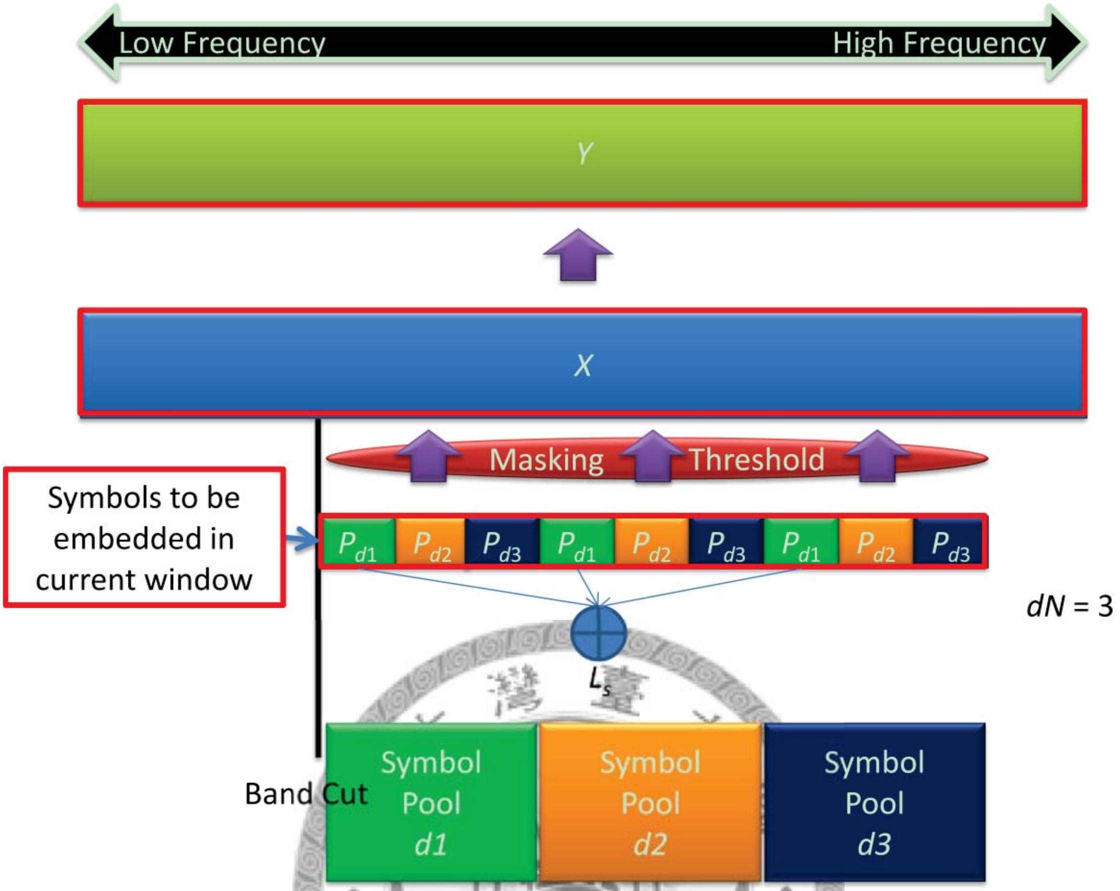

Low Frequency
High Frequency
Y
X
Masking Threshold
Symbols to be embedded in current window
P_{d1} P_{d2} P_{d3} P_{d1} P_{d2} P_{d3} P_{d1} P_{d2} P_{d3}
dN = 3
L_s
Band Cut
Symbol Pool d1
Symbol Pool d2
Symbol Pool d3

Figure 4-8. The embedding procedure of the proposed adaptive frequency division embedding, with $dN = 3$

4.3.3 Waveform-level synchronization

For a given window W in the frequency domain, the correlation between the j^{th} division and the corresponding codebook C_j is calculated by

$$\sum_{k \in \mathrm{B}_{\mathrm{dj}}} W_k \times C_{jk} \qquad (4.9)$$

The correlation within a window is the summation of the correlations of all divisions. For an offset, the correlation is the summation of all correlations of selected windows. The offset with the largest synchronization score is selected as the synchronized offset.

4.3.4 Extraction procedure

This procedure is similar to the one described in 4.2.4. In the calculation of correlation, the corresponding coefficients are interlaced instead of being sequential.

Chapter 5 Data-Level Enhancement

5.1 Overview

A variable-length code maps source symbols into bits with variable lengths. Variable-length codes allow sources to be losslessly compressed and decompressed without distortion. With the right coding strategy, a source may be compressed close to its entropy via VLC. In contrast to fixed length coding methods, VLC is more efficient in compressing data. Huffman codes, Lempel–Ziv codes and arithmetic codes are examples of well-known variable-length codes.

Most applications of the proposed system are in text-based data transmission. It needs lossless compression and the source symbols are finite, so VLC is suitable for enhancing the transmission efficiency. We adopt Huffman codes in our system.

Since the error propagation problem will cause the VLC based transmission failed, an appropriate controlling mechanism is an important factor that needs to be considered. We extend frame header structure for keeping segment-level synchronization, which specifies a specific Huffman coding area, so that the error propagation will not exceed the boundary. For this purpose, the interleaving procedure and the frame structure have to be modified the ones given in [1].

5.2 Error Propagation Caused by Huffman Codes

Huffman coding uses a specific method for choosing the representation of each symbol, and resulting in a prefix code. The bit string representing some particular symbol is never a prefix of any other symbol. For each symbol, the particular representation is stored in a Huffman table, which is derived based on the estimated probability of occurrence for each possible source symbol. The most common source symbols are represented using shorter bits than less common symbols.

For a given message consist of symbols, each symbol is first encoded into corresponding bits string in the Huffman table. The representation of the message by Huffman codes is the concatenation of these strings. For a given bits string, the decoding process uses prefix to decode symbols. Once the prefix is found to represent a symbol in the Huffman table, the symbol is decoded and the prefix is removed at the same time.

If an error occurs, prefix removing may affect the next symbol which is encoded in the string, because the length of prefix is not known. In this situation, errors are propagated to subsequent symbols. The way for controlling the error propagation is described in next section.

5.3 Controlling Mechanism of Error Propagation

In Reed-Solomon coding, the number of bits represented by a symbol is denotes as B_N. We treat f_N symbols as a frame, and collect S_N frames as a segment. In a segment, there are $S_N \times (f_N + fH_L) \times B_N$ bits, where fH_L is the length of the frame header. The number of available bits in a segment is $S_N \times f_N \times B_N$. To transmit a segment, we prepare a bit-stream which is generated by Huffman encoding. The length of the bit-stream should not exceed the number of segment available bits, and if it is less than $S_N \times f_N \times B_N$, 1's are used for padding. The Huffman table retains the symbol with all 1's to be the end symbol of the segment. See Figure 5-1 for the details.

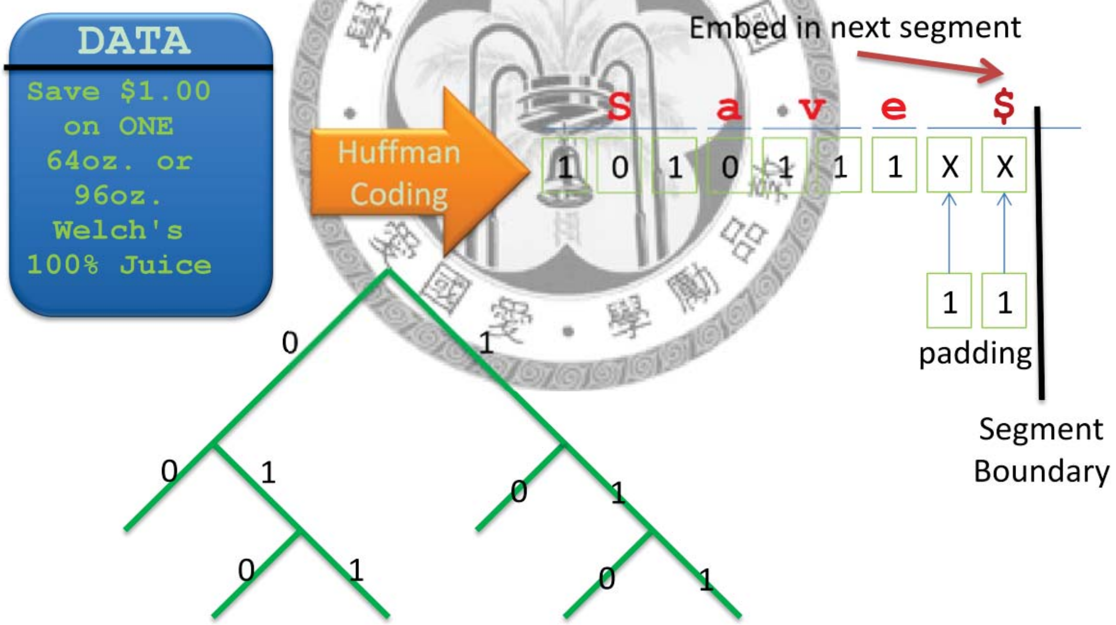
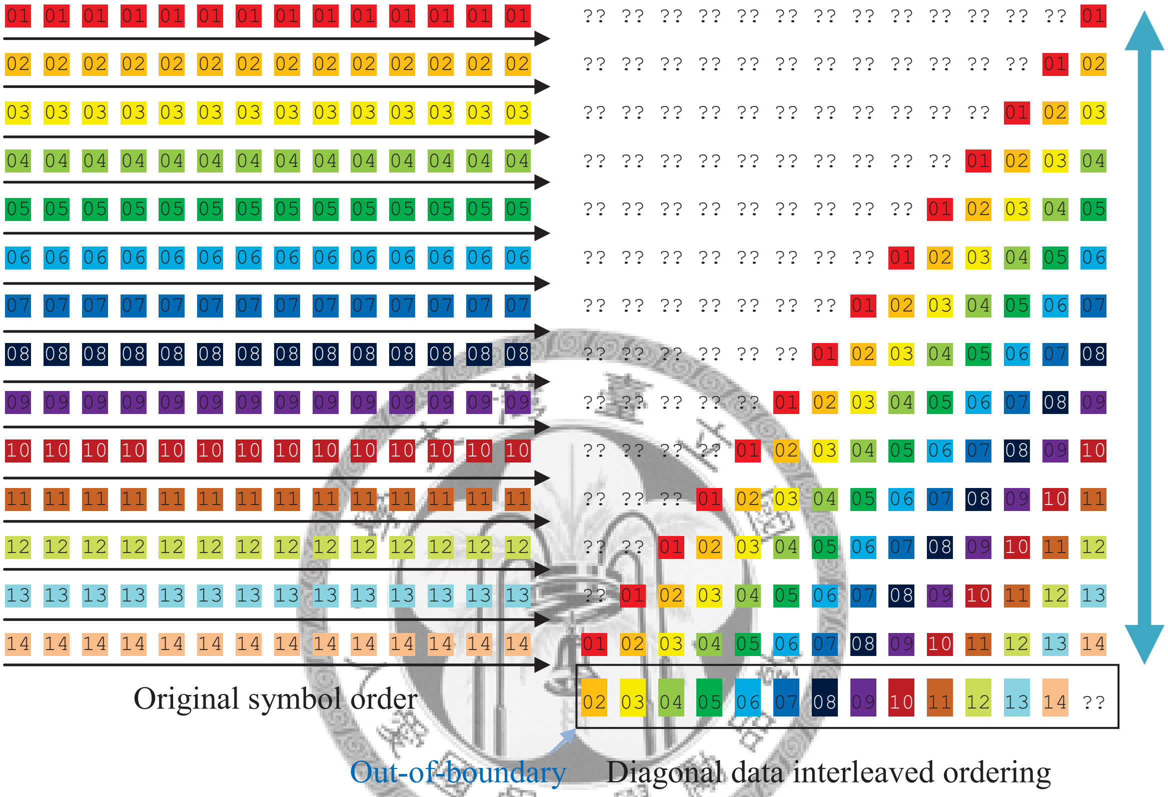


Figure 5-1. Huffman encoding

Reed-Solomon encoding procedure is used to add redundant symbols for error correction. An RS (k, n) code, which means using k symbols to represent the original n symbols. After the encoding procedure, $(S_N \times f_N) / n \times k$ symbols are obtained.

In order to resist burst errors, data interleaving is needed. Since the error propagation has to be controlled in a segment, the interleaving procedure should not make a symbol out of the segment. The diagonal data interleaver is not suitable here, please see the figure below.



We interleave symbols with the same distance, and the result is illustrated in Figure 5-2.

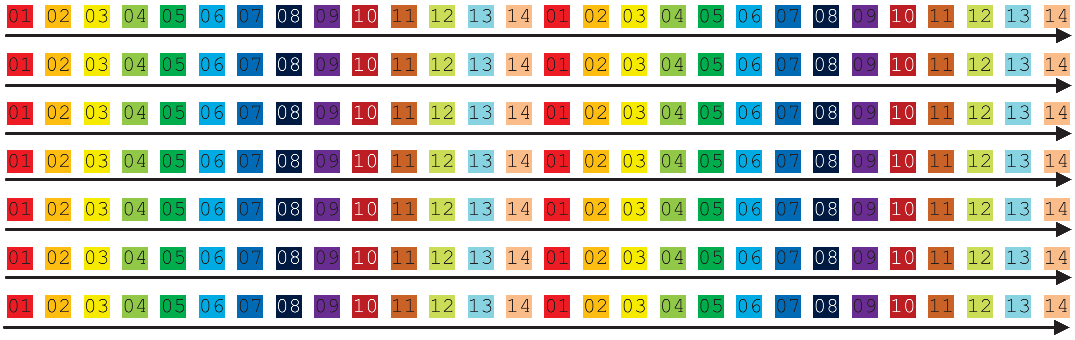


Figure 5-2. The Interleaved symbol order of the proposed data interleaver

After RS encoding and data interleaving, the frame headers are extended for maintaining the segment-level synchronization. We select two symbols with the escaping symbol to construct two types of frame headers. Figure 5-3 shows the proposed segment structure.

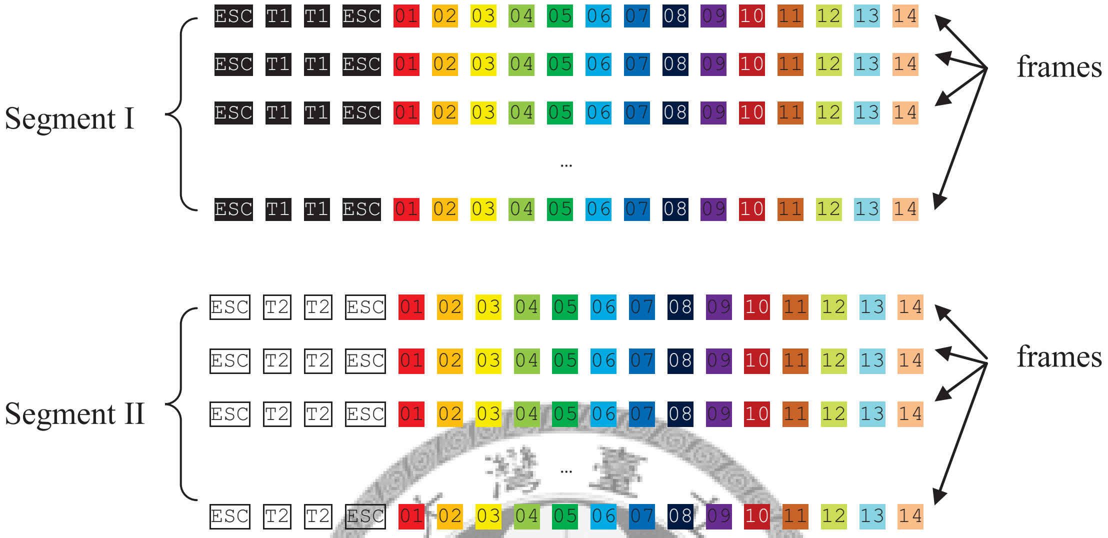
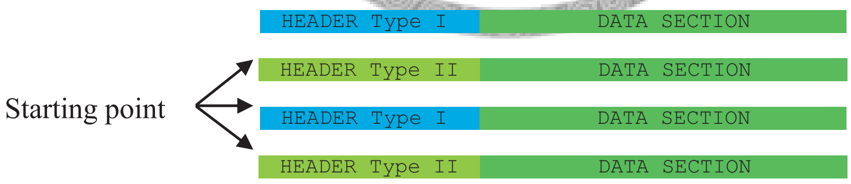

Segment I
ESC T1 T1 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
ESC T1 T1 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
ESC T1 T1 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
…
ESC T1 T1 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
frames
Segment II
ESC T2 T2 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
ESC T2 T2 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
ESC T2 T2 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
…
ESC T2 T2 ESC 01 02 03 04 05 06 07 08 09 10 11 12 13 14
frames

Figure 5-3. The proposed segment structures

A receiver can detect the changing of frame headers to recognize the starting point of an incoming segment. For example, if the frame header of the past frame is in Type I, then a starting point is detected if in this frame a Type II frame header is detected, as shown in the following figure.



Starting points of segments

Once a segment is synchronized, receiver then extracts data through inverse procedures, including de-interleaving, RS decoding, and Huffman decoding.

Chapter 6 Experiments

6.1 Environmental Configurations and System Parameters

6.1.1 Environmental configurations

For providing fair tests, we prepare almost the same environment as that proposed in [4], and the configuration of the test environment is shown in Table 6-1.

Loudspeaker	Fujitsu-Ten Eclipse TD508 II
Microphone	Sony ECM-360
DA/AD	ASUS Xonar U1

Table 6-1. The configurations of testing enviroment

6.1.2 Acoustic OFDM parameters

The parameters proposed in [5][6] are used to implement the system. Table 6-2 shows the tested OFDM system parameters. A single data frame is composed of 116 bits (36 carrier × 31 time slot). Given the coding rate 1 / 3, the effective information is 45 bytes. A data frame can be transmitted in 1.5 seconds, so the data transmission rate is approximately 240 bps.

OFDM signal band	6400 ~ 8000 Hz
OFDM symbol length	1024 samples
Guard interval / Windowing	800 samples / 224[1] samples
Number of OFDM Carriers	36
Chip rate of frame synchronization	2756 cps

Table 6-2. Acoustic OFDM system parameters

[1] In [6], 208 samples are used. The use of 224 samples leads to efficient Fourier Transform because the window length is 2048 now.

6.1.3 The parameters of the proposed approach

We implemented two frequency division approaches, as mentioned in chapter 4. Both of them are separated into 1, 2, 3, and 4 divisions, and the corresponding throughputs are 76.5625 bps, 151.125 bps, 226.6875 bps, 302.25 bps, respectively. For each entity, we examined 4 different embedding bandwidths which are all starting at 5512.5 Hz. Table 6.3 details the parameters of the proposed system.

DCT window size	512
Codebook size	16 + 1
Length of PN sequences	64, 96, 128, 160
Reed-Solomon codes	(14, 4)
Bits per RS symbol	4
Number of header types	2
Length of header in a segment	4
Length of data in a segment	14

Table 6.3. Parameters of the proposed system

6.2 Subjective Quality Test

Subjective listening tests are performed in order to evaluate the impact of data embedding. In this test, 4 different embedding bandwidths are selected in our system, and the Acoustic OFDM with parameters proposed in [6] is adopted as the comparison target.

The strength of the embedding signals is constrained by the prescribed mask, which is demonstrated on the left hand side of Figure 6-1. Another parameter which may affect the audio quality is the bandwidth for embedding signals. (See the right hand side of Figure 6-1.) The selected bandwidths are 4134.3 Hz, 5512.5 Hz, 6890.6 Hz, and 8268.75 Hz.

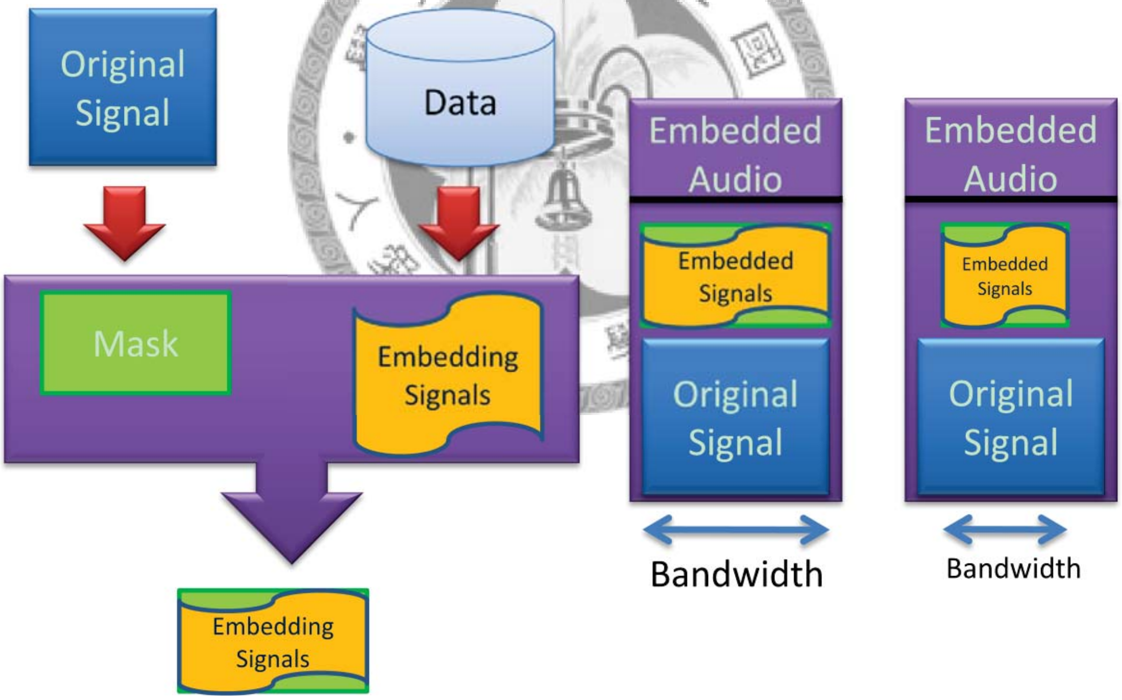


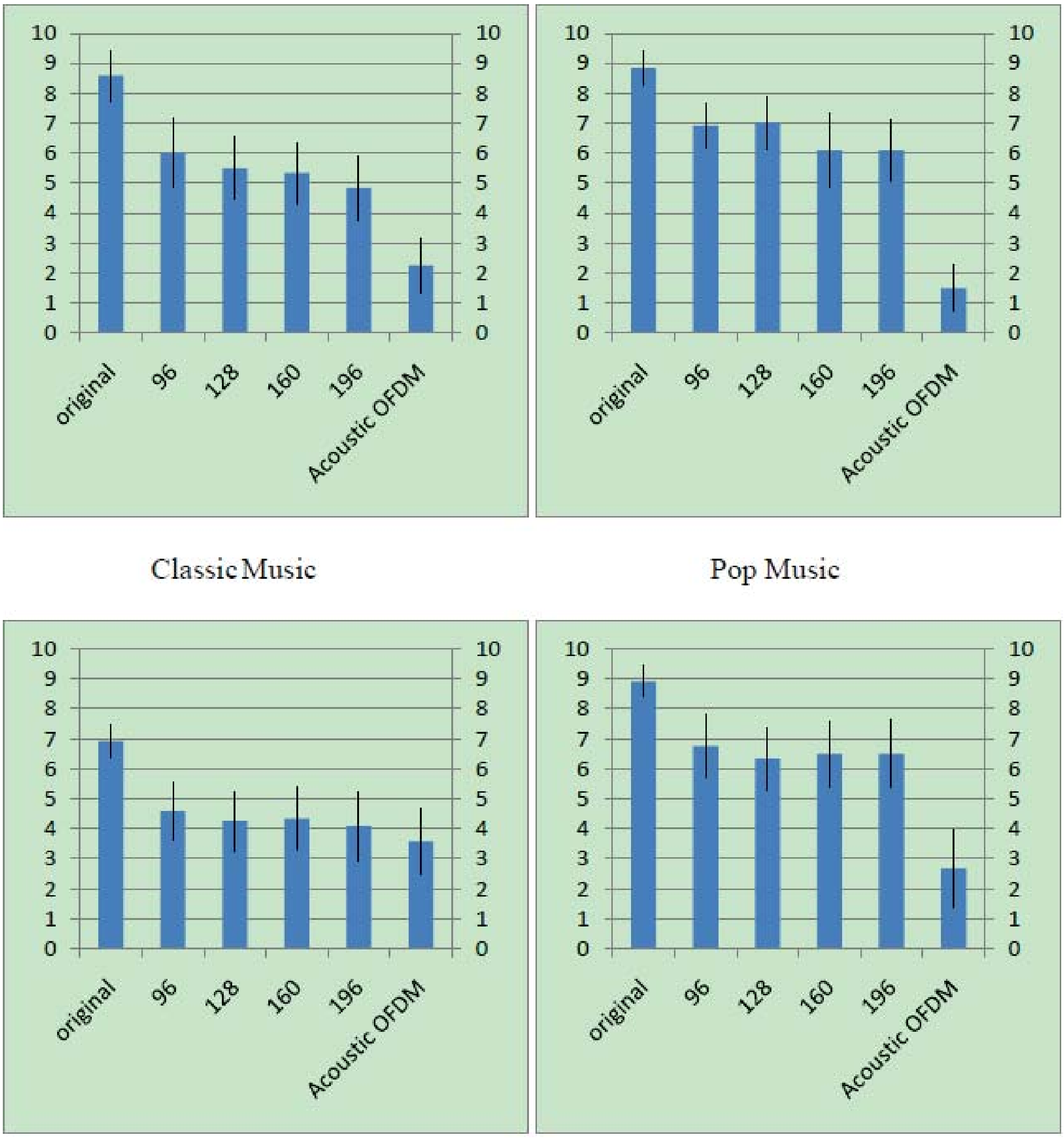
Figure 6-1. Embedding Bandwidth

In additional, different music types also make humans' feeling different. For each configuration, 4 different types of audio clips are tested, which include:

"Classic Music", "Pop Music", "Heavy Metal Music", and "Jazz Music"

Figure 6-2 shows the results obtained for each tested configuration and audio clip.

The different configurations are spread along the X-axis while the quality scale is along the Y-axis. The average over the 20 listeners is displayed, and the 95% confidence interval is for each that also shown by the vertical line segment in Figure 6-2.



Classic Music

Pop Music

Heavy Metal Music

Jazz Music

Figure 6-2. Results of the subjective test

As shown in the figure, of course, the original audio have the best quality. The qualities of SS-based method with four different embedding bandwidths are almost the same. Moreover, the results of Acoustic OFDM are not satisfactory to human ears.

For most types of music, testers can easily separate 6 audios into three parts which are the original audio, the SS-based embedded audio, and the Acoustic OFDM embedded audio. For SS-based embedding, the scores of pop music and jazz music are downgraded about 20% from the original. The classic music results, which are not as good as that of the pop music and the jazz music, may imply that people need better quality while enjoying this kind of music. The heavy metal music has the worst scores for all embedding configurations. The possible reason is that most people do not like this kind of music.

6.3 Transmission Performance

6.3.1 Factors

Besides transmission rate and transmission distance, there are still many factors or configurations which may affect the performance of transmission. For example, the playing volume, the background noise, the error rate and so on. The considering factors and the related details of each configuration are described as follows:

A. Transmission rate

In the proposed system, transmission rate can be adjusted according to *dN*, the number of divisions, as described in chapter 4. In our experiments, *dN* = 1, 2, 3, and 4, and the corresponding throughput are about 77 bps, 143 bps, 230 bps, and 307 bps. Because of the system has been integrated with VLC, the effective transmission rate for English symbols in ASCII code is equivalent to

(*Transmission Bit-rate*) × 8 / (*Average Code Length*)

We collect lots of coupon information, e-mail address and URLs from the internet. The average code length for the constructed table is about 5.4 bits per symbol, so the equivalent transmission rate is about 115 bps, 212 bps, 341 bps, and 455 bps, respectively.

B. Transmission distance

Transmission distance is determined by that measurement in the experiment. We fix other settings, and then conduct the aerial communication for keeping the distance between speaker to microphone from 0.5 meters to 6 meters stepping by

0.1 meters.

C. Volume

The playing volume is another important factor for testing the robustness of the system. Keeping other configurations the same, the higher volume can make the transmission distance longer. In this experiment, we select two sound levels, called SL-1 and SL-2, which compared to the volume in Acoustic OFDM that can transmit data successfully at distance two meters and three meters, respectively.

D. Embedding bandwidths

The embedding bandwidth affects robustness and audio quality. The wider bandwidth carries more strength of embedded signals, so it can transmit data at longer distance. The audio quality has been measured by subjective tests, and the results are shown in previous section of this chapter.

The selected bandwidths are the same as those used for subjective test, that is, 4134.3 Hz, 5512.5 Hz, 6890.6 Hz, and 8268.75 Hz respectively.

E. Error Rate

Due to text-based application, the transmission error is a serious problem which might make the whole system meaningless. In this experiment, we regard the transmission is failed if the error rate is higher than 10%. In other words, the precision of the correct data recovery in the receiver should be higher than 90%. For example, if the transmission under the configuration A can reach *x* meters with 90% precision but 89% at *x* + 0.1 meters, then the transmission distance of configuration A will be marked as *x* meters.

6.3.2 Results

A. SL1 Volume Level

Under the SL1 volume level, Acoustic OFDM can transmit data at two meter distance with throughput 240 bps and 90+% precision. Figure 6-3 and Figure 6-4 respectively show the transmission distances using different embedding bandwidths for 4 entities with different numbers of divisions, which cause different throughput rates, 77(115) bps, 143(212) bps, 230(341) bps, and 307(455) bps, respectively. In Figure 6-3, the direct frequency division is used, and in Figure 6-4, the adaptive frequency division is used.

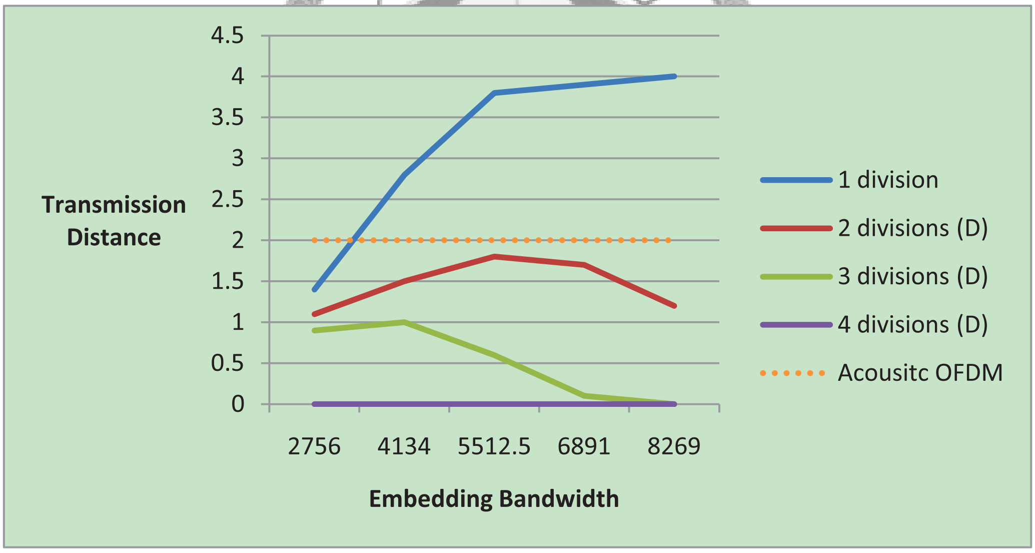


Figure 6-3. The transmission capability of the proposed expansion for SS-based method under SL1, with direct frequency division

Obviously, the transmission distance is decreasing while the bandwidth is increasing except the case of one division. Due to the selective fading, the transmission of the divisions using higher bands will be failed.

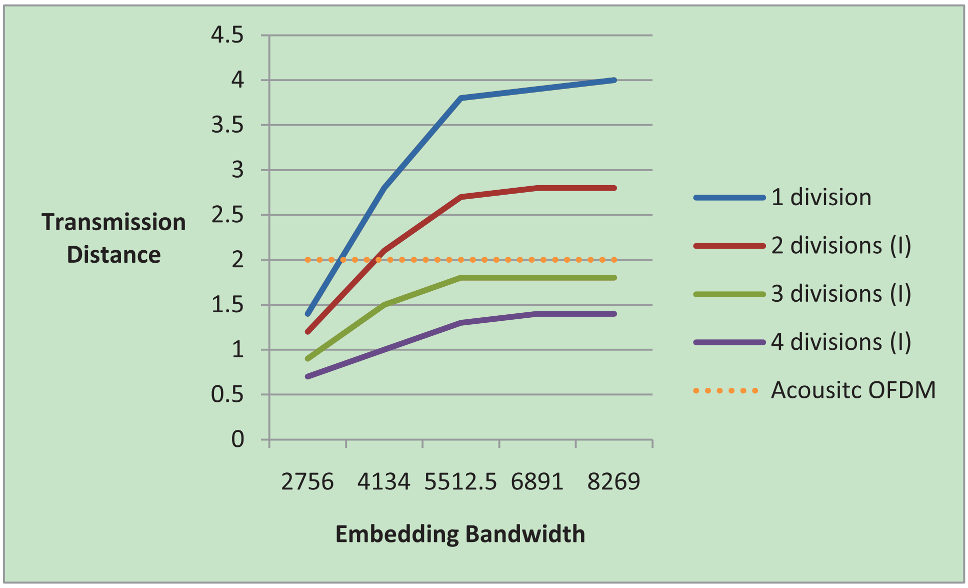


Figure 6-4. The transmission capability of the proposed expansion, for SS-based method under SL1, with adaptive frequency division

B. SL2 Volume Level

Under the volume SL2 level, Acoustic OFDM can transmit data at three meter distance with throughput 240 bps and 90+% precision. Figure 6-5 and Figure 6-6 respectively show the transmission distances using different embedding bandwidths for 4 entities with different numbers of divisions, which cause different throughput rates, say 77(115) bps, 143(212) bps, 230(341) bps, and 307(455) bps, respectively. In Figure 6-5, the direct frequency division is used, and in Figure 6-6, the adaptive frequency division is used.

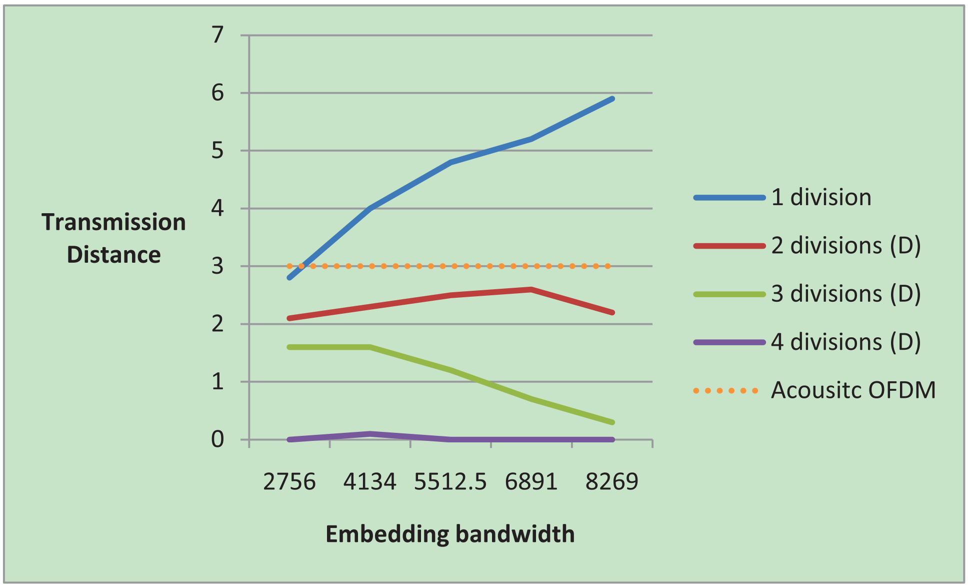


Figure 6-5. The transmission capability of the proposed expansion for SS-based method under SL2, with direct frequency division

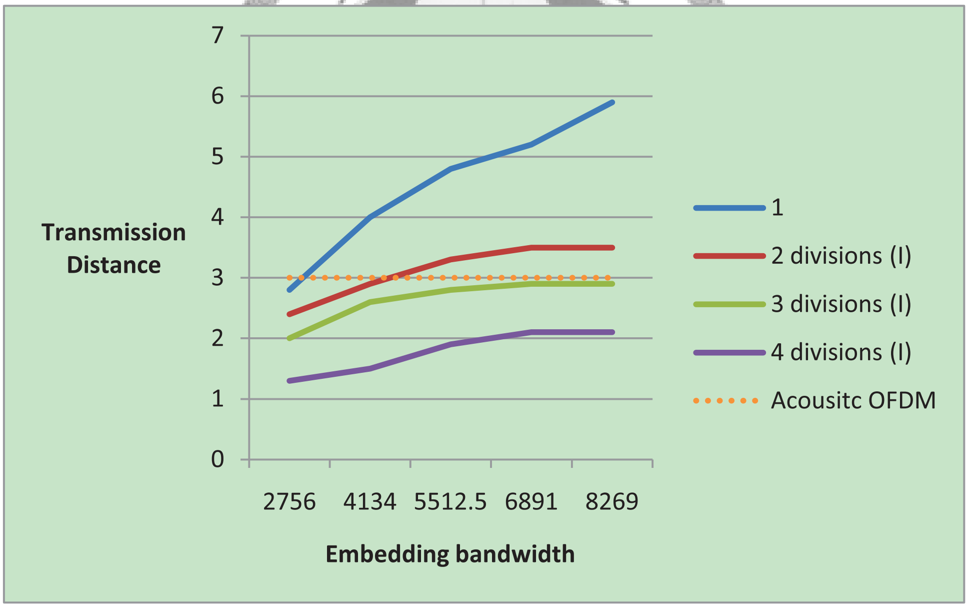


Figure 6-6. The transmission capability of the proposed expansion for SS-based method under SL2, with adaptive frequency division

Although the transmission distance is a little shorter than that of Acoustic OFDM at the similar transmission rate, the audio quality of SS-based method is much better.

Chapter 7 Conclusions and Future Work

7.1 Conclusions

In this paper, a two-level enhancement of transmission capability over the hidden aerial acoustic channel has been proposed. In the modulation level, we investigate the response of the aerial acoustic channel, and proposed an expansion to the spread spectrum based transmission systems. It extends the flexibility of the original method, and makes it possible to transmit data at high transmission rates. In the data level, the Huffman coding has been integrated into the system. It makes the transmission more effective. With nearly the same transmission distance, the enhanced SS-based method provides much better subjective audio quality as compared with the Acoustic OFDM approach.

7.2 Future Work

As mentioned in chapter 1, both SS-based method and Acoustic OFDM are state-of-the-art techniques for aerial acoustic communication. In this thesis, the improvement of SS-based embedding is achieved, but the way for improving Acoustic OFDM still under investigation, which of course, will be one of our future research topics.

Moreover, there are certain constrains for applying the proposed system in real applications, interesting research direction for resolving various constraints and/or finding tradeoff between the conflicting conditions. For instance, under a requirement of long distance and low error rate, how to achieve the highest transmission rate? Another example is to enhance the robustness under low playing volume. It may be solved by skipping some quiet windows of audios, or set the lower bound of the embedded strength.

REFERENCES

[1] Po-Wei Chen, Chun-Hsiang Huang, Yun-Chung Shen, Ja-Ling Wu, Pushing Information over Acoustic Channels. ICASSP Proc. of the 2009 IEEE Int. Conf. on Acoustics, Speech and Signal Processing, 1421-1424.

[2] Chun-Hsiang Huang, Po-Wei Chen, Ping-Yen Hsieh, and Ja-Ling Wu, "Information Delivery systems for Car Passengers without Networking Capabilities," In Proceedings of the Pacific-Rim Conference on Multimedia, pp. 539-548, December 2007.

[3] Nevena Lazic and Parham Aarabi, "Communication over an Acoustic Channel Using Data Hiding Techniques," IEEE Transactions on Multimedia, Volume 8, Issue 5, pp. 918-924, October 2006.

[4] Nakashima, Y. Matsuoka, H. Yoshimura, T. Evaluation and Demonstration of Acoustic OFDM. Signals, Systems and Computers, 2006. ACSSC '06.

[5] Matsuoka, H., Nakashima, Y., Yoshimura, T., Kawahara, T. Acoustic OFDM: embedding high bit-rate data in audio. Intl. Multimedia Modeling Conf. 2008, pp. 498-507 (2008)

[6] Matsuoka, H., Yoshimura, T., Kawahara, T. Acoustic OFDM system and its extension. In Vis. Comput(2009) 25: 3-12

[7] D. Gruhl, A. Lu and W. Bender. Echo Hiding, Information Hiding 1996, pp. 295-315

[8] Yardimei, Y., Cetin, A.E., Ansari, R. Data hiding in speech using phase coding. In Eurospeech 97, pp. 1679-1682 (September 1997)

[9] R. Hashemian, Memory efficient and high-speed search Huffman coding, IEEE Trans. on Comm., vol. 43, pp. 2576-2581, 1995.

[10] William Y. Zou, Yiyan Wu, COFDM: An Overview. IEEE Trans. on Broadcasting, vol. 41, No. 1, March 1995

[11] Teknisk Dokumentsenter, Running Fourier Transforms. Electronics World, Nov 1998

[12] L. Boney, A.H. Tewfik and K. N. Hamdy. Digital watermarks for audio signals, IEEE Ind. Conf on Multimedia Computing and System, pp473-380, Mar. 1996.

[13] ISO/IEC 11172-3:1993 Information technology -- Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s -- Part 3: Audio

[14] Jiangtao Xi and Joe F. Chicharo, "Computing Running DCTs and DSTs Based on Their Second-Order Shift Properties," IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications, Volume 47, Issue 5, pp. 779-783, May 2000.

[15] Stephen B. Wicker and Vijay K. Bhargava, "Reed-Solomon Codes and Their Applications," IEEE Press, ISBN 0-7803-5391-9, September 1999.