Sound Tone Communication

MTP Stage I Report

Submitted in partial fulfillment of requirements for the degree of

Master of Technology

by

Nisha

Roll No.-143050072

under the guidance of

Prof. Bhaskaran Raman

Department of Computer Science and Engineering

Indian Institute of Technology, Bombay

October, 17 2015

Abstract

This report presents analysis and results of the implementation of audio based networking. Audio based networking uses acoustic channel as medium and can be used for short-range communication with low data rates. For the implementation of sound tone communication, the inaudible frequency range (16KHz to 18.5KHz) is used. We have given a synchronisation mechanism for sender and receiver by using preamble and a start symbol. Two error correction techniques are implemented to avoid back channel. For physical layer design, three different modulation schemes - ASK, FSK and 4-ary FSK are implemented and analysed. The analysis of these modulation schemes is given on the basis of throughput i.e how throughput will vary according to parameters - start symbol length, data bit length and number of data bits in the frame.

Contents

1 Introduction **5**

2 Previous Work **5**

3 Motivation **6**

4 Definitions **6**

5 Physical Layer Design Parameters **7**
- 5.1 Frequency Selection 7
- 5.2 Modulation Techniques 8
- 5.3 Sender: Physical Layer 9
- 5.4 Receiver: Physical Layer 10
- 5.5 Synchronisation Mechanism 10
  - 5.5.1 Preamble 10
  - 5.5.2 Start Symbol 11
- 5.6 Error Correction 12

6 Performance Evolution **13**
- 6.1 Experiment Setup 13
- 6.2 Results 13
  - 6.2.1 Frequency Shift Keying (FSK) 13
  - 6.2.2 4-ary Frequency Shift Keying (4-FSK) 16
  - 6.2.3 Amplitude Shift Keying (ASK) 16

7 Conclusion and Future Work **19**
- 7.1 Conclusion 19
- 7.2 Future Work 19

List of Figures

1 Frequency signal power level vs SNR 8
2 Sender at Physical Layer 9
3 Receiver at Physical Layer 10
4 Throughput with and without start symbol 12
5 Throughput analysis: FSK 14
6 Throughput vs Data Bits per Frame 15
7 4-FSK:Throughput vs Data Bits per Frame 16
8 Throughput analysis: ASK 17
9 ASK:Throughput vs Data Bits per Frame 18

List of Tables

1 Packet Loss rate vs Data bits per frame 15
2 ASK: Error rate vs Data bits per frame 18

1 Introduction

Today everyone has smart phone and they need to transfer of information for communication purpose. Wireless technologies such as WiFi, Bluetooth, 3G etc. guarantee communication speed and accuracy, but they require special hardware installed on phone which increases their cost, and all radio wave based communications also consumes high power. Sound tone communication is an audio based networking, which uses acoustic channel as medium for transmission. In this, data is modulated over different sound frequencies up to 22.1KHz(Nyquist limit), and then transmitted over air using speakers of the device. The receiver, record the sound waves using microphone and decode the actual data. Therefore, it do not require any additional hardware, hence, it is completely software based approach. One advantage of sound tone communication over other radio wave communication technologies is that is is not connection-oriented, therefore there is no time overhead of connection setup.

Smartphones are able to sample audio at the rate of 44.1K, therefore, a smartphone can generate sound up to 22.1KHz according to Nyquist limmit. This frequency range can be divided into two parts- audible and inaudible sound waves. The 20Hz to 15KHz is audible range of human beings and above that is inaudible. In acoustic channel the modulation schemes can be same as that used for EM wave based communication technologies, such as, Amplitude shift keying(ASK), Frequency shift keying(FSK), Phase shift keying(PSK), M-ary frequency shift keying(MFSK), OFDM etc..

2 Previous Work

In recent past, applications are developed for small data transfer in short-range with audible as well as inaudible sound waves. [1] and [2] given context aware computing applications using audible sound waves at up to 251bps at distance of 30cm and using 21KHz achieved rate of 8bps at distance of 3.4m. Dhwani [3], used audible frequency(6KHz-7KHz) to implement NFC. It implements OFDM technique and achieved data rate of 2.4Kbps in less than 20cm range. Prewhisper [4], similar to dhawani, implements short-range communication by using 9KHz carrier and MFSK modulation and achieved data rate of 1kbps in range of less then 0.5cm. A system for Audio signalling based NAT traversal [5] , used FSK modulation scheme over 1200Hz-1300Hz frequencies and achieved data rate up to 8bps within few centimeters range. Chirpcast [6], implemented audio based communication within range of 1m-2m using inaudible frequencies 18KHz-22.1KHz and using FSK modulation scheme it achieved data rate of 4bps. Chirp Signal-Based Aerial Acoustic Communication for Smart Devices [7] uses chirp BOK modulation scheme over frequency range of 19.5KHz to 22KHz and achieved data rate of 16bps in 25m range. On Covert Acoustical Mesh

Networks in Air [8] used already existing network stack for underwater communication systems over 21KHz frequency, and achieved data rate of 20bps up to 25m range. Only [4] and [5] implemented error correction mechanism at physical layer.

From this survey we can say that the short-range communication will give higher data rate for small distance communication and as distance increased data rate will start decreasing. By analyzing all the experimental data in above papers, it can be concluded that the frequency range that can be used for audible communication is from 8KHz to 12KHz and that for inaudible communication 16KHz to 22KHz.

3 Motivation

This work is part of project CARTS, Communication Assisted Road Transport System [9], which looks at the problem of road traffic congestion in developing regions from a broader perspective. While traveling in a bus, to get higher accuracy in GPS location and to save battery life of smart phone, the users can share their GPS location via smart phones. As GPS location detection is itself high power consuming, we can use sound based communication to share GPS location between commuters within a bus.
So for above purpose we need a data transmission scheme that can transfer small data in short-range(i.e. up to 5m) and robust to high noise, as traffic on roads is very noisy. As the application is going to be used in public place, therefore, we can use only inaudible range of frequencies i.e. 16KHz to 22KHz, so that it should not disturb other passengers on the way.
In previous work, using inaudible frequencies the maximum data rate achieved is 16bps at distance upto 25m with 19.5KHz to 22KHz, and other had achieved 4bps to 8bps within distance of 1m to 3m. As of my best knowledge till the date no work is done for the noisy environment using inaudible frequencies. In MTP phase I, we designed and implemented physical layer of sound tone communication using three different modulation techniques and analysed the results on different design parameters.

4 Definitions

- **Loss Rate:** A packet is lost when at receiver side there is no preamble detected in the packet. The loss rate is given by the following formula:

$$Loss\ rate = \frac{Number\,of\,lost\,frames}{Total\,frames\,sent}$$

- **Error Rate:** A frame is in error when received data is not same as the data transmitted by sender i.e a frame will be a error frame if even a single bit is in error. The error rate is given by following formula:

$$Error\ rate = \frac{Number\,of\,Frames\,in\,error}{Total\,frames\,sent - Lost\,frames}$$

- **Throughput:** Throughput is the overall data rate of the transmission while considering the error rate as well as the loss rate, as the actual data rate may reduce due to errors and packet loss. The formula for throughput is given as:

$$Throughput = \frac{Number\,of\,frames\,not\,in\,error * bits\,per\,frame}{Total\,frames\,sent * one\,frame\,duration}$$

- **FFT:** Fast Fourier transform used to convert a signal from time domain to frequency domain. By FFT, we are finding the maximum componet of a frequency in given number of samples. In our experiments to keep simple and less complex FFT we used only samples in form of 2^N. FFT algorithm we followed is part of "Project Nayuki,2014 at MIT to compute the discrete fourier transformation of a vector" [10]. This implementation uses the Cooley-Tukey decimation-in-time radix-2 algorithm and its complexity is $O(nlogn)$.

5 Physical Layer Design Parameters

In this section, all the parameters of physical layer implementation of this appliction are described. The medium of transmission is air and the speakers and microphones of communication devices works as sender and receiver respectively.

5.1 Frequency Selection

An analysis is done on frequencies in range from 16KHz-22KHz to get the optimal renge of frequencies that can be used for this application. As the goal is to use this application in bus, so we conducted experiments in city buses to identify the operatable frequencies. First we conducted an experiment in silence up to 8 meter distance and found out that frequencies greater than or equal to 20KHz are not separable from noise as their power level is very low. Then we used frequencies 16KHz, 17KHz, 18KHz, 18.5KHz, 19KHz in outdoor environment i.e. buses to check their power levels in such high noise.
For Experiment we used android application *Frequency generator* to generate different kind of frequencies in Sony Xperia Z3 and recorded the sound using android application *Easy Recorder* in Samsung Grand. We conducted 5 experiments for each frequency at 5 different distances and used *Auda City* software to analyse the power levels of recorded data. The graph 1 represents the power levels of signals and corresponding SNR.

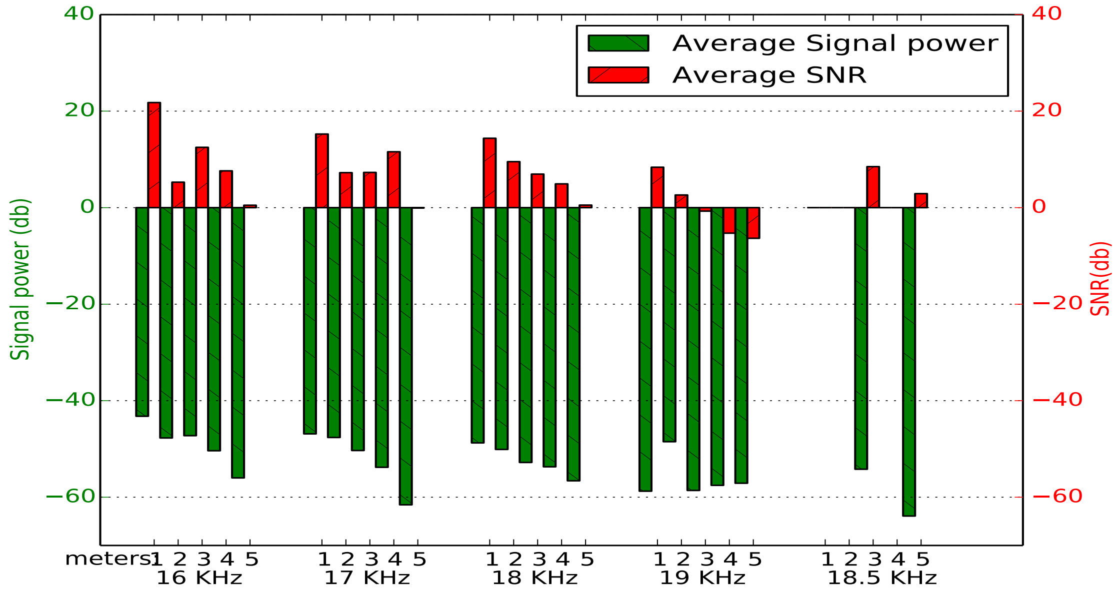

Figure 1: Frequency signal power level vs SNR

This graph shows that 19KHz frequency can not be detectable in high noisy environment, as its SNR is very low at large distance and it is not seperable from noise. Therefor, we excluded the use of 19KHz. After that, we also checked 18.5KHz at distance 3m and 5m, which given a good SNR. Now, the range of frequencies that can be used for this application is from *16KHz to 18.5KHz*, i.e. below 19KHz.

5.2 Modulation Techniques

The modulation techniques that are used for EM waves based communication(e.g. WiFi, Bluetooth) can also be used for audio based communication. For this application we worked on following modulation schemes:

1. **Amplitude Shift Keying (ASK):** In this scheme the binary bit '1' is represented by a frequency and the bit '0' is absence of that frequency. It is a simply On-Off keying (OOK) with a carrier frequency. In our experiments we used 17KHz for representing bit '1' and bit '0' is absence of 17KHz.

2. **Frequency Shift Keying (FSK):** In this modulation scheme, the different bits or symbols are represented by different frequencies. This scheme can be divided into two parts

- **Binary Frequency Shift Keying(BFSK):** In this scheme, two binary bits are differentiated based on two different frequencies. Any pair of frequencies between 16KHz-18.5KHz can be used to implement BFSK. In our experiments we used 17KHz to represent bit '0' and 18.5KHz to represent bit '1'.
- **4-ary Frequency Shift Keying(4FSK):** This scheme uses four different frequencies to transmit 4 symbols. This is a case of MFSK where for M number of symbols M different frequencies are used as carrier. For our experiments we selected frequencies such that the difference between two frequencies is atlest 500Hz. The frequencies used for our implementation are:

Symbol	Frequency
00	16.5KHz
01	17.0KHz
10	18.5KHz
11	17.5KHz

5.3 Sender: Physical Layer

This section describes the design of sender at physical layer. First of all, the data bits are encoded using encoder. Now, data bits and the preamble are modulated according to their respective modulation schemes and start symbol is added. Now this whole frame is transmitted in air via speakers. The figure 2 shows the implementation design of sender.

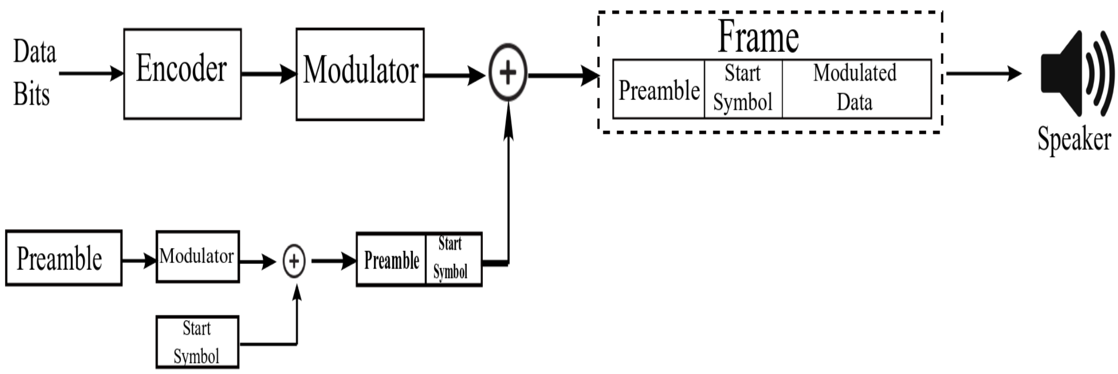

Figure 2: Sender at Physical Layer

5.4 Receiver: Physical Layer

This section describes the receiver's working at physical layer. The transmitted waves are recorded using microphone and then synchronisation mechanism is applied over the recorded data. After applying preamble detection and start symbol detection we will get the start of the data frame. Now, after removing preamble and start symbol from the frame we left with only modulated data bits. Now this data is demodulated by applying FFT for each bit duration and according to the frequency of those samples the value of bit is decided either '0' or '1' depending on modulation scheme. After demodulation, we have encoded bits and now decoder will decode these bits and gives the actual data bits. The figure 3 shows working of the receiver at physical layer.

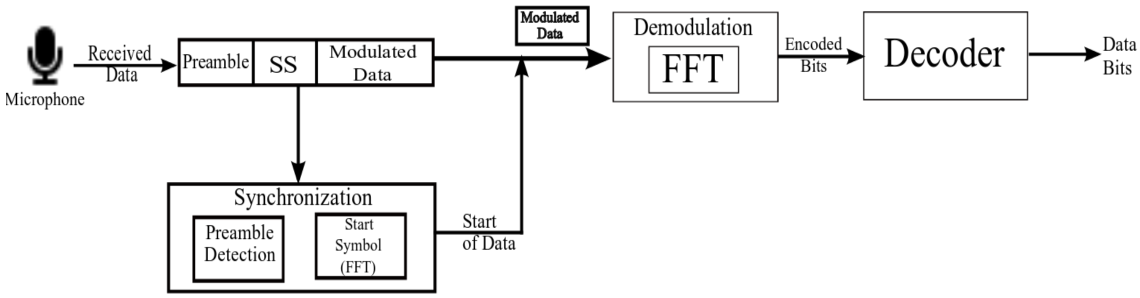

Figure 3: Receiver at Physical Layer

5.5 Synchronisation Mechanism

5.5.1 Preamble

For any communication system the most important part is to identify the start of the frame or data. That means there must be some synchronisation between sender and receiver. For this synchronisation purpose a preamble is used to detect the start of frame. Sender first generate a preamble by replicating first half as the second half and modulate it over the carrier frequency. Now at receiver, first the received data is quantized so that it will reduce effect of by filtering. A window of N samples is maintained, where N is the number of samples in preamble. *Pearson's auto-correlation* is applied on two halves of sliding window and on this correlation data, peak detection algorithm is applied to find out where the correlation between two halves of window peaks and that will be the start of frame.

The pearson's auto-correlation formula that used is:

$$correalation(x,y) = \frac{\sum_{i=1}^{p/2} x_i y_i}{\sqrt{\sum_{i=1}^{p/2} x_i{}^2 * \sum_{i=1}^{p/2} y_i{}^2}}$$

The peak detection algorithm uses EWMA(Exponentially Weighted Moving Average) to get higher accuracy in peak positions. The maximum peak values is considered as the start of frame.

The selection of preamble is very tedious task and therefor we tested several combinations of preambles to choose best one. First we set the duration of preamble 1 second, i.e. enough duration to get start with. Then we tested several randome generated preambles(5 bits to 110 bits) using ASK as well as BFSK. On the basis of results shown in [11], we got the best preamble of fixed 110 bits for duration 0.5sec with ASK modulation scheme. The one half of the preamble is:

$$1010010011011111000000100010001110111010110010100101101 \quad (1)$$

5.5.2 Start Symbol

Now, after sending data with preamble, it is observed that the correlation may peak in data part also and it will lead to multiple peaks and may result in false peak detection's from previous approach. Therefore, we introduced a *Start Symbol* after preamble. Now, after every peak it will check for start symbol and when it detects the start symbol, that peak is accurate and will be the start of frame. The start symbol is of 18KHz and receiver uses FFT to detect the frequency of it.

The rate of successful reception of frame is increased from 7% to 75.2%, but at the same time it may seems as overhead to use extra bits for start symbol. So, we compared the throughput of transmission with different length start symbols and that is shown in figure 4

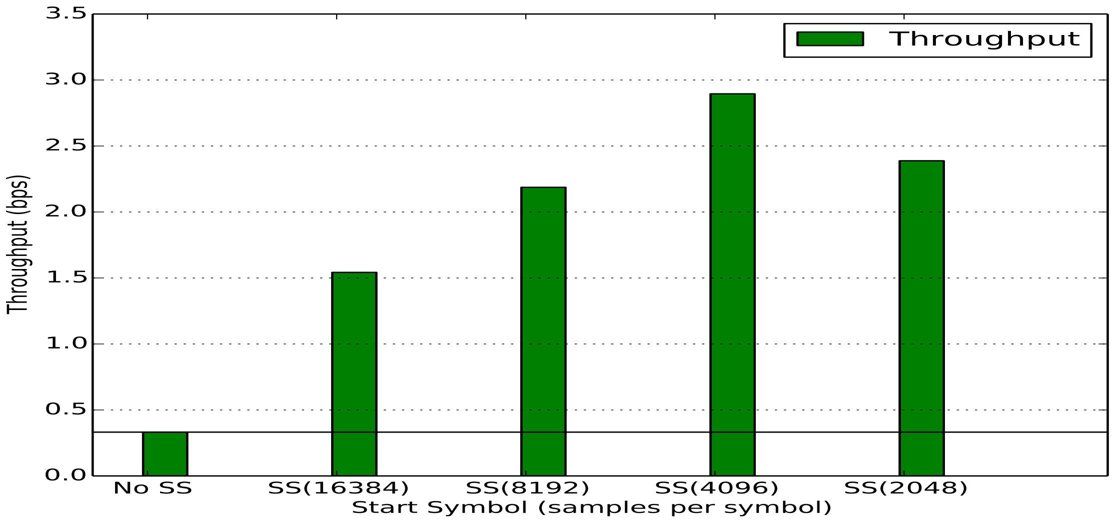

Figure 4: Throughput with and without start symbol

This figure shows that even by using different lengths of start symbol the throughput is always higher than the throughput of transmissions without a start symbols.
So, the advantages we get by introducing a start symbol after preamble are:

- It will eliminate the false peak detection problem
- As the preamble detection is not always at exact position, we can use start symbol to shift position to get more accuracy in start of data in frame.

Now, after detecting start symbol, the start of data in frame is calculated by adding preamble length and start symbol length in peak position, and then next data is fed to modulator for further processing.

5.6 Error Correction

Data is the field where actual data bits starts in frame. The data bits are encoded, therefore, at receiver a decoder will decode the data bits and obtains actual data. As the data bits may corrupt due to noise, we decided to use an error correction mechanism. We implemented and analysed two coding techniques under *Forward Error Correction*.

- **Hamming Codes:** Hamming codes are block codes which calculate parity bits over a block of code. We used (7,4,3) hamming code for encoding implementation i.e. total of 7 bits includes 4 bits of data and 3 bits of parity. It is able to correct up to one bit error.

- **Viterbi Codes:** This is a convolution encoding scheme in which original data bits never sent over channel. Two bits are generated for each bit position in data bits by applying bit operations, hence, it will generate the encoded bits twice the original data bits. It is able to correct up to two bit errors.

The detailed encoding decoding mechanism is described in [11]. The results of both encoding schemes are also compared in [11].

6 Performance Evolution

To decide a perticular modulation scheme and packet length, we experimented over different parameters for each of three modulation techniques described above. This section shows the performance of each modulation scheme and their comparison.

6.1 Experiment Setup

Our target devices are smartphones, but to decide over so many parameters and for the purpose of analysing data we used laptops as sending and receiving devices. Sender is DELL-studio and receiver is Lenovo-G450. The distance between sender and receiver is 2 meter and all the experiments conducted in regular environment. Results are presented on basis of 500 transmissions for each perameter set. For synchronisation and automation of sender and receiver we used TCP sockets.

6.2 Results

6.2.1 Frequency Shift Keying (FSK)

In this section, FSK modulation scheme is tested on different lengths of start symbol and data bit. All these results are with respect to implements viterbi encoding scheme. First, we considered start symbol as well as data bits of length 16384 and then decreased samples numbers from both in power of 2, as we are using FFT which works for number of samples which are in form of 2^n. We always kept the duration of start symbol greater than or equal to the duration of data bits. There are total 15 such combinations of start symbol duration and data bit duration. For each combination, there are four data bits in a frame. The graph 5 shows the throughput achieved for each combination.

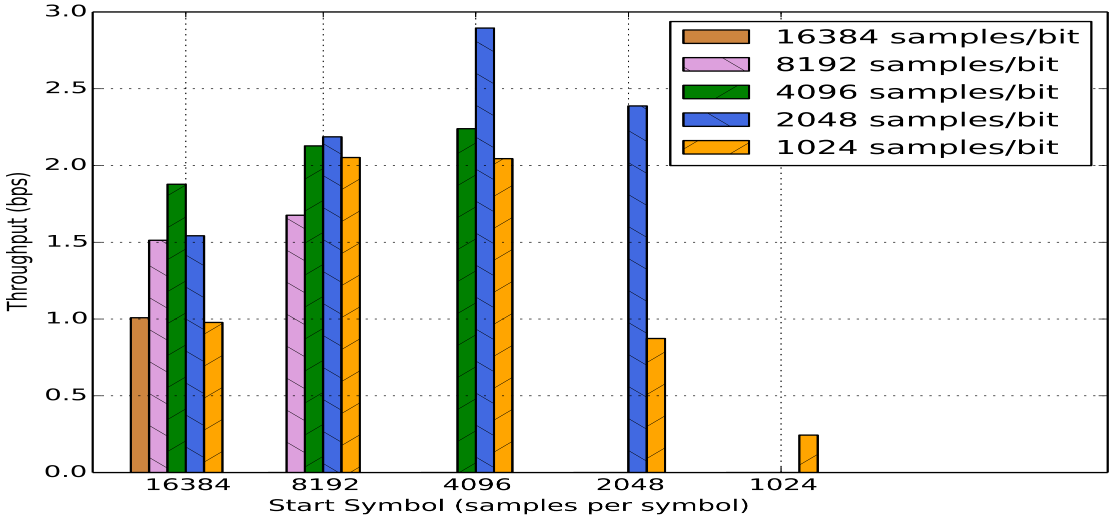

Figure 5: Throughput analysis: FSK

It is observed from the error analysis for FSK done in [11], that there is a continuous increase in error rate while decreasing the bit-duration for each start symbol. In graph 5,Initialy, throughput is low even for very less error rate (i.e.¡10%) because the bit overhead is more. Now, when we fix the start symbol duration the throughput increases as the bit-duration decreas, but after bit-duration 4096 there is a drop in throughput and reason for this is that, for small start symbol and small bit-duration the error rate is very high and also the packet loss rate increased for small start symbol. This same behavior can be seen for each of the start symbol in above graph5. So, the throughput will be good for the combination where start symbol and bit-duration overhead is low and at the same time the error rate is less than 30%.

From above results, we have selected two best combinations i.e. 4096-2048 and 2048-2048, which are giving the highest throughput. Now for these two combinations we increased the number of data bits in frame. The following graph6 shows the variation in throughput by increasing the number of data bits in the frame.

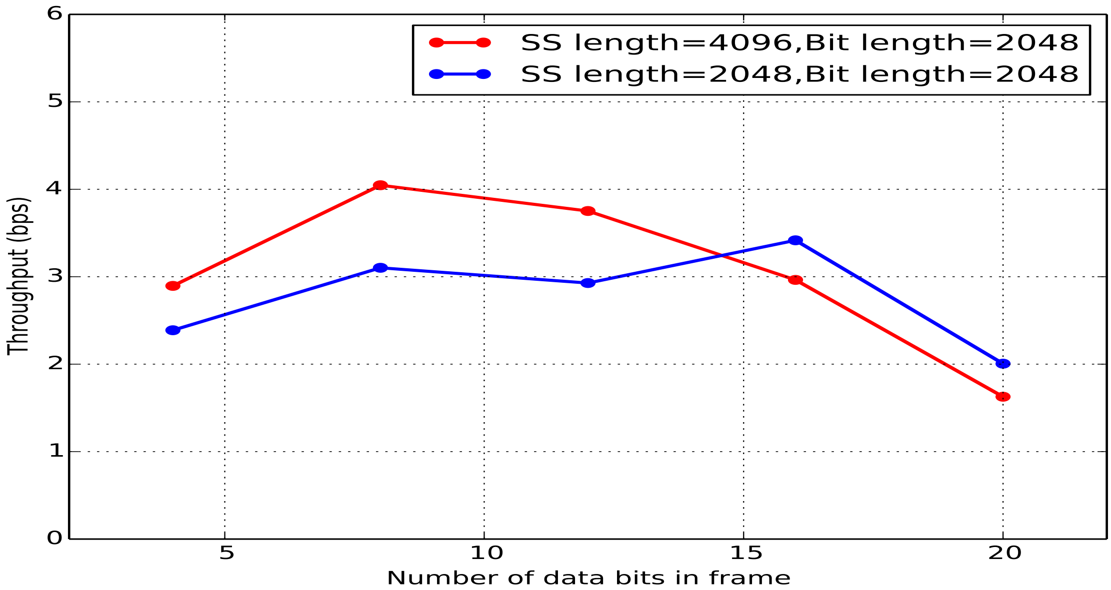

Figure 6: Throughput vs Data Bits per Frame

Here we can see that when number of data bits increases in frame, the throughput will increase, but after a certain payload length (8 bits) it starts to drop. Again by increasing data bits, error rate in received data increases as analysed in [11] and therefore the throughput will also decreases. Here, error rate will play major role in variation of throughput not loss rate, because it is observed that for a particular combination of start-symbol and data-bit duration the loss rate remains almost consistent as shown in table 1

Number of Bits	8192-8192	4096-2048	2048-2048
4	5.5	7.2	23.6
8	7.7	7.2	22
12	5.088062622	6.2	23.94736842
16	8.695652174	10.6	18.4

Table 1: Packet Loss rate vs Data bits per frame

There is continuous drop in throughput for the combination 4096-2048 after 8 data bits per frame, because the error rate increased, and for the combination 2048-2048, the error rate is increasing slowly, therefore the throughput will remain consistent for the certain frame length.

6.2.2 4-ary Frequency Shift Keying (4-FSK)

The analysis of 4-FSK is also according to viterbi encoding scheme. The error rate behavior for 4-FSK is very similar to that of for FSK i.e. there is continuous increase in error rate while increasing the number of bits in the frame. From the results of FSK modulation, we selected two best combination of start symbol and data bit duration and then varied the number of data bits in packet. The graph 7 shows the variation in throughput when data bits increased per frame.

Here, the throughput continue to increase till 12 bits per frame and then there is a little

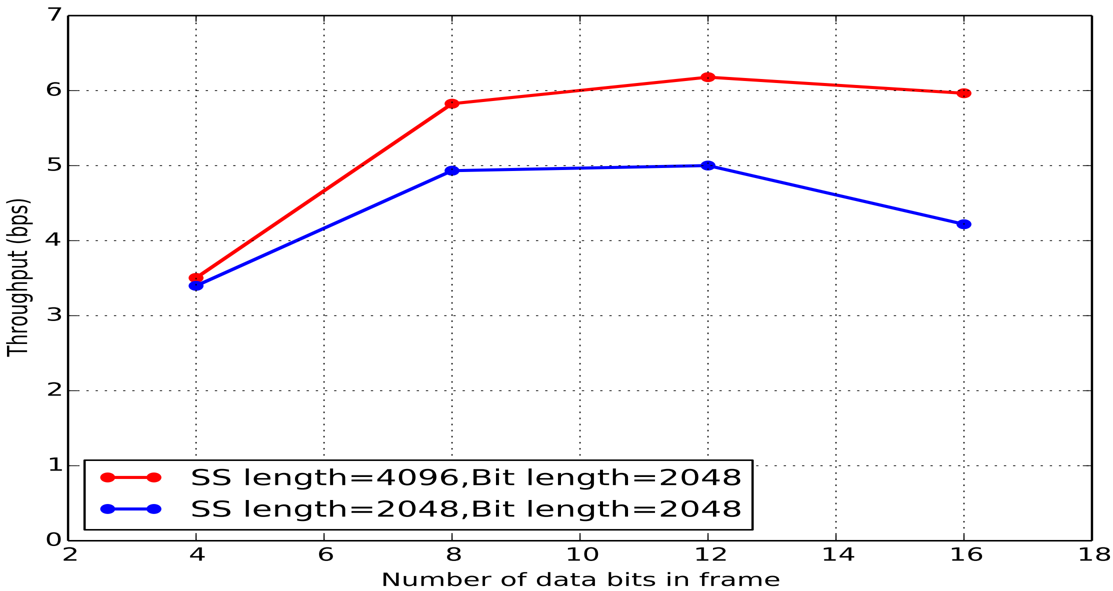

Figure 7: 4-FSK:Throughput vs Data Bits per Frame

drop because the error rate continue to increase. Throughputs are higher compared to FSK, as there are 2 bits in one symbol.

6.2.3 Amplitude Shift Keying (ASK)

For ASK we did analysis for possible combinations of start symbol and bit durations. Hamming encoding is used for implementing and testing ASK modulation. The behavior of ASK modulation is very different from that of FSK and 4-FSK, as it is very sensitive to the environment noise. For ASK we do not considered the sample length 16384 as it is overhead even for very small error rates. The graph 8 shows throughput on different combination of start symbol and data bits duration.

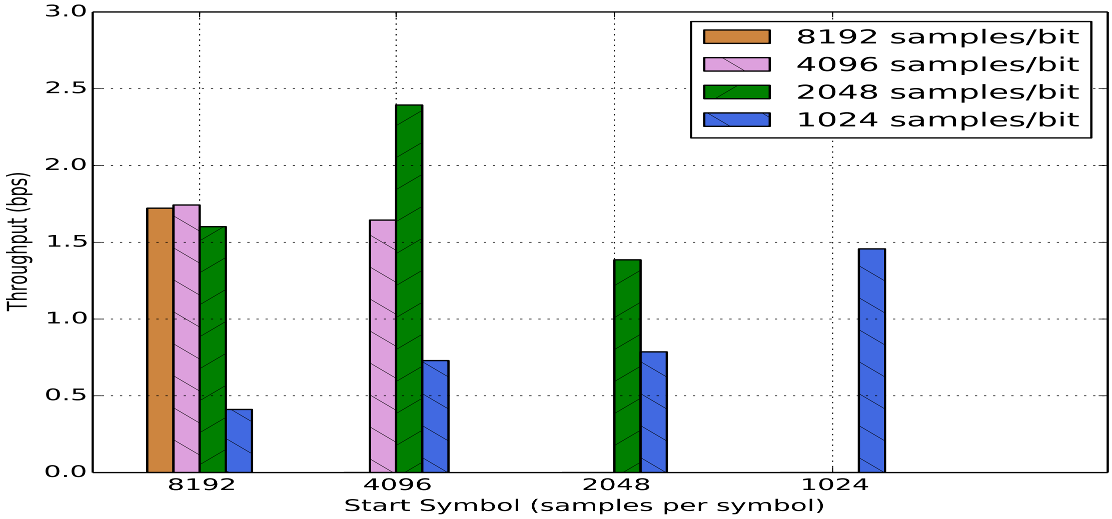

Figure 8: Throughput analysis: ASK

The throughput behavior for a fixed start-symbol is same as in FSK i.e initially increases because bit-duration overhead in decreases and the error rate increases slowly. But after bit-duration 4096 there is a drop in throughput because for small bit-duration error rate increased to more than 60%. For small start-symbol also the packet loss increased and error rate also increased, hence the throughput is low. Now, from above graph, we selected two best throughput resulting combinations of start-symbol and data-bit duration which are: 8192-8192 and 4096-2048, and then varied the data bits per frame.

The graph 9 shows the variation in throughput when data bits per frame are increased.

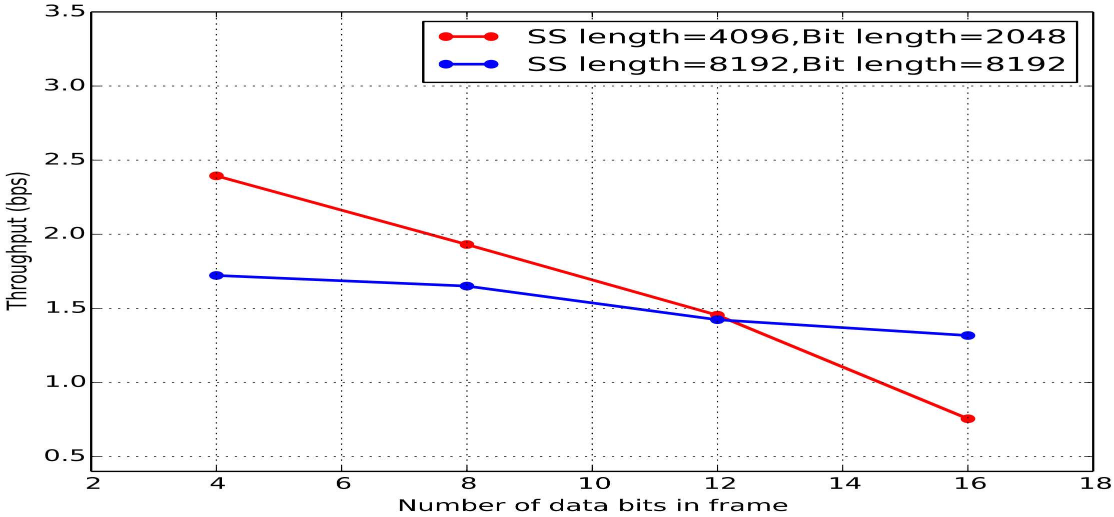

Figure 9: ASK:Throughput vs Data Bits per Frame

The behavior of both the combination is same as the data bits per frame increased the throughput will continue to decrease. The reason for this behavior of ASK is that there is a high increment in the error rate when data bits per frame are increased. This high increment in error is shown in table 2.

Number of Bits	8192-8192	4096-2048
4	9.523809524	36.75496689
8	26.54387866	67.03296703
12	42.68041237	80.04201681
16	46.93877551	90

Table 2: ASK: Error rate vs Data bits per frame

Throughput is maximum for the combination 4096-2048 as the bit-duration overhead is low. Here also, there is not much effect of loss rate on throughput because, for the same combination loss rate remains consistent. The throughputs are lower than that of in FSK because the error rates for ASK are very high.

7 Conclusion and Future Work

7.1 Conclusion

From above analysis we get different throughput from different modulation techniques. In FSK, the maximum throughput we achieved is *4.05 bps* when start symbol is of 4096 samples, bits are of 2048 samples each and there are 8 data bits in one frame. In 4-FSK, the maximum throughput achieved is *6.18 bps* when start symbol is of 4096 samples, bits are of 2048 samples each, and there are 12 data bits in one frame. In the literature survey, I did not find any implementation or analysis of ASK modulation with 16KHz-18.5KHz frequencies for smartphones, therefore, we also implemented and analysed the behavior of ASK modulation technique. The maximum throughput achieved for ASK is *2.394 bps* when start-symbol is of 4096 samples, bits are of 2048 samples, and one frame consists of 4 data bits only. I do not find ASK as a good modulation scheme that can be used in audio based networking, as it is very sensitive to environmental noise.

We achieved a throughput of 6 bps for the frequency range 16KHz-18.5KHz using 4-FSK modulation scheme, which is maximum in all the previous works on same range of frequencies with these modulation schemes.

7.2 Future Work

1. To analyse the same implementation in noisy environment and also with varying distance
2. Implementation and analysis of modulation scheme 8FSK
3. Implementation of link layer and media access control layer
4. Implementation and analysis using smartphone in actual environment

Acknowledgment

I would like to thank Prof. Bhaskaran Raman for the guidance and valuable feedback he have provided, without which this work would not have been possible.

References

[1] A. Madhavapeddy, D. Scott, A. Tse, and R. Sharp, "Audio networking: The forgotten wireless technology," 2005.

[2] A. Madhavapeddy, D. Scott, and R. Sharp, "Context-Aware Computing with Sound," *Proceedings of The International Conference on Ubiquitous Computing (UbiComp '03)*, pp. 315–332, 2003.

[3] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan, "Dhwani: Secure Peer-to-Peer Acoustic NFC," in *ACM SIGCOMM*, pp. 63–74, 2013.

[4] B. Zhang, Q. Zhan, S. Chen, M. Li, K. Ren, C. Wang, and D. Ma, "PriWhisper: Enabling Keyless Secure Acoustic Communication for Smartphones," *IEEE Internet of Things Journal*, vol. PP, no. 1, pp. 1–1, 2014.

[5] A. Patro, Y. Ma, F. Panahi, J. Walker, and S. Banerjee, "A system for Audio signalling based NAT traversal," in *2011 3rd International Conference on Communication Systems and Networks, COMSNETS 2011*, 2011.

[6] F. Iannacci and Y. Huang, "ChirpCast : Data Transmission via Audio," *University of Washington*, 2010.

[7] H. Lee, T. H. Kim, J. W. Choi, and S. Choi, "Chirp Signal-Based Aerial Acoustic Communication for Smart Devices," in *International conference INFOCOM 2015*, 2014.

[8] M. Hanspach and M. Goetz, "On Covert Acoustical Mesh Networks in Air - Volume 8, No. 11, November 2013 - Journal of Communications," *Journal of Communications,*, vol. 8, pp. 758–767, 2013.

[9] B. Raman, "Carts: Communication assisted road transportation systems." `http://www.cse.iitb.ac.in/silmaril/br/doku.php?id=proj:carts`, 2008.

[10] P. Nayuki, "Free fft and convolution." `http://www.nayuki.io/res/free-small-fft-in-multiple-languages/Fft.java`, 2014.

[11] Mahima, "Sound tone communication." `http://www.cse.iitb.ac.in/synerg/doku.php?id=public:students:mahima:home`, 2015.