# LPC Compression Channel Exploration

Allison Crim

## Introduction

Linear predictive coding or LPC is used in a variety of ways in the field of audio signal processing, specifically in speech processing. In this project, LPC is used to compress an audio signal to the bare essentials to transmit over a communication channel. This report will explore the tradeoffs for various parameters to reduce the amount of data that needs to be sent over the communication channel. The report will also explore methods for voice detection and pitch detection.

## Linear Predictive Coding

Linear predictive coding uses knowledge of how speech is produced to break down audio signals into a vocal tract filter and a source that models the vocal folds. The vocal tract filter can be complex or simple and for this project, the vocal tract filter is assumed to be an all-pole filter. The coefficients for this filter are fixed and can be found using Levinson-Durbin recursion. In this project the matlab function *lpc* was used to quickly find these coefficients. The *lpc* matlab command also returns the gain of the system.

The other component of LPC that needs to be understood is the source that models the vocal folds. Depending on whether the audio signal is voiced or unvoiced, the source will either be an impulse train or random noise. A voiced signal will be a uniform periodic train at pitch period P and an unvoiced signal will be flat spectrum noise. The spectrum of the source is assumed to be flat for simplicity. A diagram of LPC is shown below.

Diagram

Description automatically generated

*Figure 1: LPC Compression Diagram*

The result of the LPC compression for this project is an array of data that indicates the gain, vocal tract filter coefficients, and whether the signal is voiced or unvoiced. The ideal circumstance of the compression is to have the fewest filter coefficients that produces the best quality resynthesized signal. Several parameters can be adjusted in this compression scheme to tune the output. These parameters will be explored in the next section.

## Parameter Experimentation Results

The two parameters that were varied in this project to improve quality were frame size and filter order (number of vocal tract coefficients). The resulting signals were overall most clear when the filter order was 100 or more but the signals were legible even when the filter order was less than 20. For most of the signals I discovered that a frame size of 50 ms sounded the best. The graph below shows a comparison of a signal with 100 vocal tract coefficients versus a signal with 15 vocal tract coefficients.

A picture containing timeline

Description automatically generated

*Figure 2: Comparison of Changing Number of Coefficients*

It is not extremely clear visually that the signal with 100 vocal tract coefficients sounds better but it is clear from listening. The one major difference that is noticeable in the above graph is that the top graph has a larger amplitude which could be why the signal sounds much better.

## Changing Parameters for Specific Signal

The next portion of this project focused on optimizing a specific signal rather than trying to do a general case. The sentence that was chosen is ‘The synthesized signal is supposed to be of high standard.’ This sentence is tricky because it contains a lot of fricatives specifically the ‘s’ sound. It appears that through experimentation with the code, increasing the number of vocal tract coefficients and keeping the frame size at 50 ms gave the best tradeoff of quality and amount of data being sent over.

Graphical user interface

Description automatically generated with low confidence

*Figure 3: Comparison of Input Signal and Resynthesized Signal*

## Conclusion

This project explored the effect of changing the parameters of frame size and number of vocal tract coefficients in an LPC compression system. In most areas of communication systems it is common for there to be a tradeoff between quality of the signal and amount of data being sent and in this case it is no different. In the evaluation of the parameters using different input signals, it was clear that a frame size of 50 ms was ideal and that the best number of coefficients for getting a signal that was clear enough to understand was between 20 and 50. If the end user desired a signal that sounds very close to the signal that was inputted into the system, the ideal number of coefficients is closer to 75. Linear predictive coding is a useful tool for breaking down large signals into manageable bites to send across a communication channel.