Digital Voice Codec: LPC Implementation

Technical Case Study & System Design

Melik Ayberk Türkeli
B.Sc. Candidate, Electrical & Electronics Engineering
Boğaziçi University, Istanbul

Abstract

This technical paper details the design and implementation of a Linear Predictive Coding (LPC) vocoder for low-bitrate speech coding. Addressing the bandwidth constraints of modern communication channels, the developed system leverages the source-filter model of speech production to achieve significant compression ratios while maintaining intelligibility. The project encompasses the full DSP pipeline, including signal pre-processing, Levinson-Durbin recursion for spectral envelope estimation, pitch detection algorithms, and excitation synthesis. Implemented in Python, the system demonstrates proficiency in digital signal processing, algorithmic optimization, and audio analysis.

**Note: This project was originally developed at Boğaziçi University within the scope of the EE473 Digital Signal Processing course.*

1 Problem Definition & Objectives

Modern communication systems demand highly efficient utilization of bandwidth and storage, especially when transmitting speech over constrained or shared channels. Traditional waveform-based coding methods, such as Pulse Code Modulation (PCM), operate at high bitrates, consuming substantial channel capacity and creating bottlenecks in delay-sensitive applications like VoIP and satellite links.

To address these constraints, this study focuses on producing intelligible speech at low bitrates without relying on direct waveform transmission. Unlike waveform coders that represent the signal sample-by-sample, this project adopts a model-based approach. By exploiting the physical structure of human speech production, the system shifts toward an analytical representation of speech.

The primary objective is the design and implementation of a Linear Predictive Coding (LPC) vocoder. This system reinterprets speech as the output of a physical production model consisting of an excitation source (vocal folds) and a spectral shaping filter (vocal tract). By transmitting only model parameters—LPC coefficients, gain, pitch, and voicing information—the vocoder significantly reduces the required bitrate. The resulting system is assessed in terms of compression efficiency, reconstructed speech quality, and algorithmic stability.

2 Technical Background

Human speech can be mathematically modeled as the output of a source-filter structure [2]. In this model, the vocal folds generate an excitation signal, which is subsequently shaped by the vocal tract into recognizable phonetic units. Linear Predictive Coding (LPC) is derived from this perspective, modeling each speech sample as a weighted sum of previous samples

[1]. This enables a compact representation of spectral characteristics by transmitting prediction coefficients instead of the raw waveform.

While modern neural vocoders (e.g., LPCNet, WaveNet) offer superior naturalness, traditional LPC remains the foundational framework for understanding parametric speech coding. It provides a transparent analytical model that clearly demonstrates the trade-off between bitrate and perceptual quality [4].

3 System Architecture & Methodology

The proposed system is implemented as an analysis-synthesis pipeline operating on short-time frames. This frame-based approach allows the system to track the time-varying nature of speech while maintaining computational efficiency.

3.1 Linear Prediction Model

In LPC analysis, the vocal tract is modeled as an all-pole filter with the transfer function:

$$H(z) = \frac{G}{1 - \sum_{k=1}^{p} a_k z^{-k}}, \tag{1}$$

where a_k denote the linear prediction coefficients and G represents the gain. The prediction error corresponds to the excitation signal. The coefficients are determined by minimizing the mean squared prediction error over each frame using the autocorrelation method.

3.2 LPC Coefficient Estimation

The optimal prediction coefficients are derived by solving the Yule-Walker equations:

$$\sum_{k=1}^{p} a_k R[|i-k|] = R[i], \quad i = 1, 2, \dots, p, \tag{2}$$

This system is solved efficiently using the **Levinson-Durbin recursion**, which exploits the Toeplitz structure of the autocorrelation matrix. An open-source reference implementation of the recursion algorithm was adapted and optimized for this project to ensure numerical stability.

3.3 Pitch Estimation & V/UV Decision

A critical component of the encoder is the classification of frames as Voiced or Unvoiced (V/UV) and the estimation of pitch:

- **V/UV Decision:** Based on short-time energy, zero-crossing rate (ZCR), and autocorrelation peaks.
- **Pitch Estimation:** For voiced frames, the fundamental period T_0 is estimated by maximizing the autocorrelation function within the human pitch range:

$$T_0 = \arg \max_{\tau_{\min} \le \tau \le \tau_{\max}} R_{xx}[\tau]. \tag{3}$$

3.4 Speech Synthesis (Decoder)

The decoder reconstructs the speech signal using the transmitted parameters. The excitation signal is generated synthetically:

$$e[n] = \begin{cases} \sum_k \delta[n - kT_0], & \text{voiced (Impulse Train)} \\ \mathcal{N}(0, \sigma^2), & \text{unvoiced (White Noise).} \end{cases} \tag{4}$$

This excitation drives the LPC synthesis filter. To ensure seamless reconstruction, an **Overlap-Add** method is employed, mitigating frame boundary artifacts.

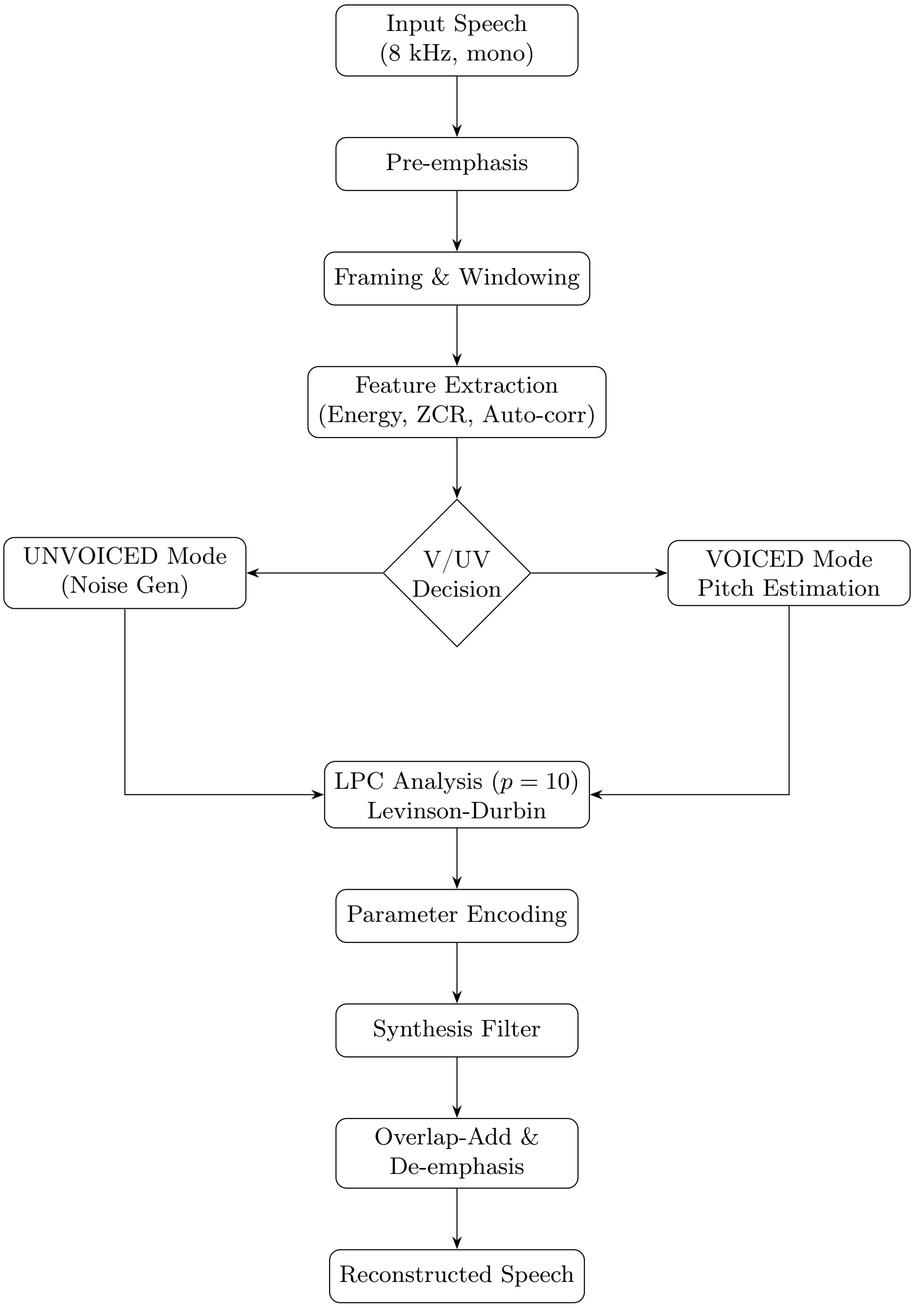
Input Speech (8 kHz, mono)
Pre-emphasis
Framing & Windowing
Feature Extraction (Energy, ZCR, Auto-corr)
V/UV Decision
UNVOICED Mode (Noise Gen)
VOICED Mode Pitch Estimation
LPC Analysis ($p = 10$) Levinson-Durbin
Parameter Encoding
Synthesis Filter
Overlap-Add & De-emphasis
Reconstructed Speech

Figure 1: System Flowchart: From Analysis to Synthesis.

4 Implementation & Challenges

The system was implemented in Python using the `NumPy` and `SciPy` libraries. The development process encountered and resolved several engineering challenges:

- **Filter Stability:** Synthesis filters occasionally became unstable when LPC poles approached the unit circle. This was mitigated by applying *bandwidth expansion* ($\gamma \approx 0.98$) to the coefficients.
- **Pitch Tracking:** Low-energy voiced frames caused ambiguity in pitch detection. A reliability threshold based on normalized autocorrelation was introduced to robustly distinguish these from unvoiced segments.
- **Artifact Removal:** Frame discontinuities were resolved using Hamming windows during analysis and strict overlap-add reconstruction during synthesis.

5 Performance Evaluation

The system was evaluated using standard speech processing datasets. The encoder achieved a theoretical bitrate of approximately **9.3 kbps**, representing a compression ratio of **13.8:1** compared to standard 128 kbps PCM.

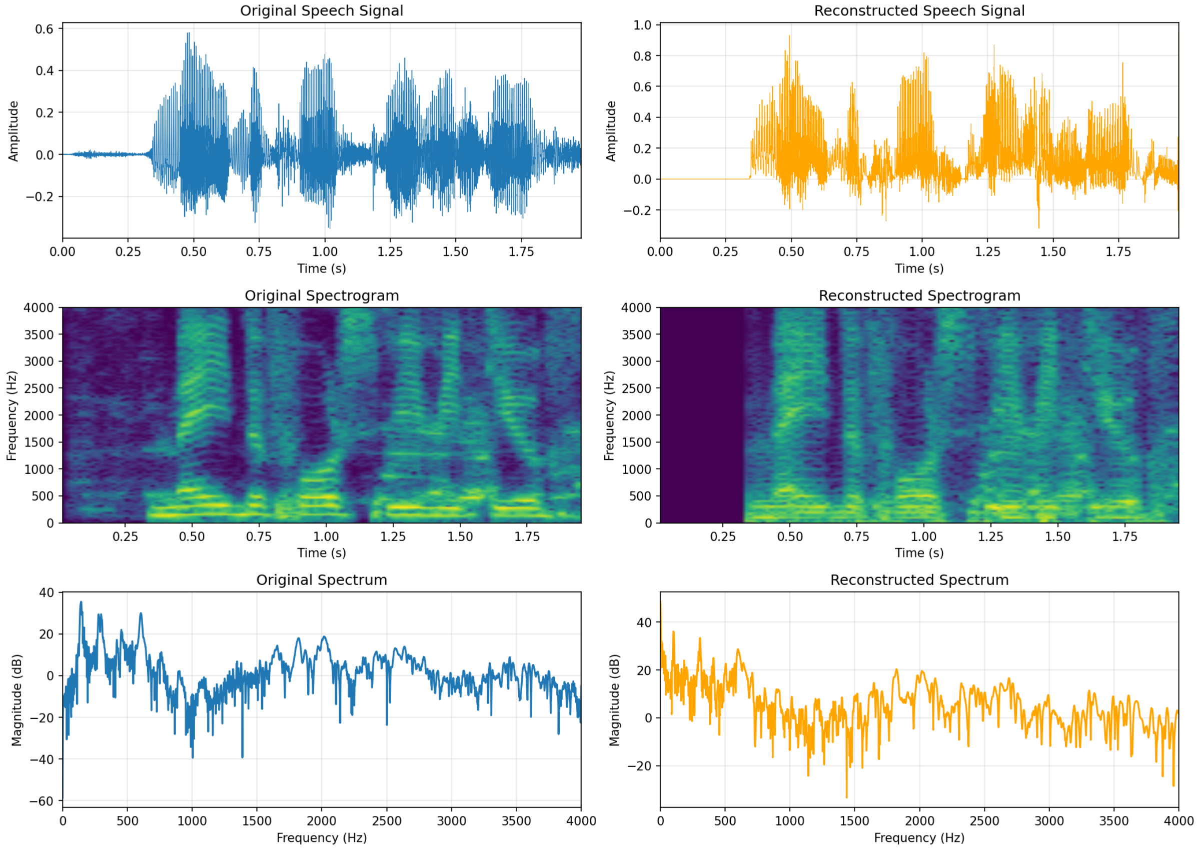

Figure 2: Time and Frequency domain comparison. Left: Original Signal. Right: Reconstructed Signal showing preservation of the spectral envelope.

Qualitative analysis (Figure 2) confirms that while fine harmonic details are simplified (resulting in a slightly robotic timbre), the spectral envelope and formant structures are preserved, ensuring high intelligibility.

6 Conclusion

This case study demonstrates a successful implementation of a parametric speech coding system. By abstracting the speech waveform into model parameters, the system achieves substantial bandwidth reduction suitable for constrained channels. The project highlights the practical application of digital signal processing theories and provides a robust foundation for advanced vocoder designs such as CELP or neural synthesis.

7 References

1. Atal, B. S., and Hanauer, S. L. (1971). Speech analysis and synthesis by linear prediction of the speech wave. *J. Acoust. Soc. Am.*

2. Rabiner, L. R., and Schafer, R. W. (1978). *Digital Processing of Speech Signals.*

3. Oppenheim, A. V., and Schafer, R. W. (1999). *Discrete-Time Signal Processing.*

4. Deller, J. R., et al. (2000). *Discrete-Time Processing of Speech Signals.*