

Pole-Based Source-Filter Speech Reconstruction: A Prototype and Bitstream Audit

Daniel Schmitter

Abstract

A speech representation based on excitation and a low-order resonant filter can be much smaller than a waveform, but a compact synthesis model is not yet a low-bitrate codec. We analyze a source-filter prototype motivated by operator-based signal representations. Its framewise least-squares recurrence, pole stabilization, voiced/unvoiced excitation, and persistent synthesis state are described explicitly and related to classical linear prediction. Four deterministic synthetic fixtures verify finite stable frame filters, with a 150 Hz sinusoid assigned approximately 150.94 Hz. Source inspection reveals that the reported nominal bitrates are not obtained from a serialized stream, that the in-memory voiced payload omits an energy quantity used directly by the synthesizer, and that a related vector-quantization benchmark learns and evaluates its codebook on the same recording. Thirteen binary64 filter coefficients alone would occupy 104 bytes per frame, or 33,280 bits/s at the stated frame duration, before pitch, gain, and framing. The study therefore documents a parametric reconstruction mechanism and reproducible qualification failures, not a new speech codec, validated intelligibility result, or demonstrated spline-specific compression advantage.

1 Introduction

Speech has structure that a sample-by-sample representation does not expose. A periodic or noisy excitation passing through a resonant filter can explain important aspects of voiced and unvoiced sound. This suggests a small representation whose parameters have physical interpretations. The engineering challenge is to transmit those parameters reliably and reconstruct useful speech at a measured bit rate, not merely to generate recognizable sound inside one process.

We examine an existing pole-based reconstruction prototype and its related codebook benchmark. The purpose is to separate the demonstrated numerical mechanism from unsupported codec claims. We provide a self-contained description, elementary stability analysis, deterministic signal fixtures, and payload accounting. The implementation is rooted in classical source-filter and linear-prediction methods; a new spline-specific contribution or competitive rate-quality result is not established.

2 Relation to linear prediction and neural codecs

Linear prediction models a signal through past samples and an excitation, with an all-pole spectrum as an important case [1]. The recurrence below and its least-squares estimation are instances of that framework. Calling its polynomial roots operator poles does not by itself distinguish it from established LPC analysis. Likewise, a persistent IIR filter state is not evidence of a new continuous-time spline synthesis theorem.

Modern neural codecs such as SoundStream jointly learn compact codes and reconstruction systems and evaluate a complete compression pipeline [2]. Our prototype contains no comparable trained neural encoder/decoder, entropy model, or held-out listening evaluation. It is motivated by the same resource question but operates at a different evidence level. A future comparison would need actual encoded bytes and task-relevant quality, not an assigned target rate.

3 Architecture and information flow

The source-filter architecture has three different state objects: frame parameters, persistent synthesis state, and the original recording used during analysis. A valid codec may transmit the first and evolve the second, but the decoder cannot consult the third. In the prototype, voiced synthesis receives an original energy value absent from its nominal message, so the illustrated process is not yet an independently executable decoder.

The decoder is the test of compression



Read left to right above, then right to left below. Arrows show data or computation flow.

Figure 1: The decoder is the test of compression. Boxes distinguish supplied information, fitted components, and the quantity evaluated. Arrows show computation or data dependence, not a newly trained deep network.

Stable poles ensure a useful local numerical property, but they do not establish perceptual quality or a complete rate budget. Pitch, gain, voicing, codebook identity, and frame synchronization all cross the interface. A codebook trained on the same recording also carries information about the evaluation material. Counting only its index omits that acquisition and storage dependency.

4 Frame analysis and operator representation

The implementation samples at 16 kHz and uses nonoverlapping 25 ms frames of 400 samples. It pre-emphasizes the waveform by subtracting 0.97 times the preceding sample. Voicing and pitch are estimated from a second-order 600 Hz low-pass filtered frame using normalized autocorrelation over lags corresponding approximately to 50-400 Hz. A peak exceeding 0.45 produces a voiced decision. These are fixed heuristics, not validated universal physiological thresholds.

$$e_n = x_n + \sum_{j=1}^p a_j x_{n-j}, \quad A(z) = 1 + \sum_{j=1}^p a_j z^{-j}, \quad (1)$$

$$\hat{a} = \arg \min_a \|Xa + y\|_2^2, \quad H(z) = 1/A(z). \quad (2)$$

For voiced frames, the order-twelve regression uses the Hamming-windowed waveform. For unvoiced frames it instead regresses a segment of the autocorrelation. The leading filter coefficient is one. The code checks roots of the denominator polynomial. If any root lies outside the unit disk, it radially normalizes all roots, not only the offending root, and reconstructs a real polynomial. It then multiplies coefficient j by 0.95 to power j .

For a monic degree- p denominator polynomial, coefficient scaling by gamma to power j replaces every root r by gamma times r . Thus the final bandwidth expansion contracts roots. In exact arithmetic, roots already inside the disk become strictly stable for gamma below one, while the normalization branch moves their moduli below one before contraction. Floating root finding and coefficient reconstruction still require numerical checking, especially at higher order. Stable individual filters also do not automatically guarantee uniformly bounded behavior under arbitrary rapid coefficient switching.

5 Synthesis state and transmitted information

Voiced excitation uses a periodic impulse train with a phase offset carried across frames, a 3 kHz low-pass shaping filter, and added Gaussian noise. Unvoiced excitation is Gaussian noise. Excitation amplitudes use the original frame energy. The all-pole synthesis carries a filter state between frames, followed by de-emphasis

after concatenation. These mechanisms generate a parametric waveform; they do not reproduce the original excitation samples.

A decoder separated from the encoder must receive or deterministically reconstruct every quantity on which that waveform depends. The in-memory payload contains a voicing flag, a field holding pitch for voiced frames or energy for unvoiced frames, and the full coefficient vector. In the same-process demonstration, the synthesizer also receives the original energy directly. For voiced frames, that energy is absent from the stated payload. Consequently, the payload is not by itself a complete decoder interface.

$$R_{\text{payload}} = \frac{b_{\text{envelope}} + b_{\text{pitch}} + b_{\text{gain}} + b_{\text{voicing}} + b_{\text{framing}}}{0.025 \text{ s}}. \quad (3)$$

The script declares a twelve-bit target constant but does not use it to quantize or serialize the parameters. Its final message assumes approximately 48 bits per frame. Neither number is a measurement. At binary64 precision, the thirteen coefficients alone occupy 104 bytes per frame, corresponding to 33,280 bits/s before other fields. Even omitting the known leading coefficient leaves 30,720 bits/s of coefficient payload. These arithmetic figures are not the actual size of Python dictionaries, nor a claim that full-precision coefficients are necessary for a future codec.

6 Experimental methods

The audit reads the complete frame-analysis and synthesis source and the related automatic benchmark. It leaves both unchanged. To avoid microphone access, playback, and dependence on an installed audio driver, the validation script extracts only the deterministic pitch and pole-analysis function definitions from the source syntax tree. It supplies NumPy, SciPy signal operations, and the original sample rate and frame length explicitly.

Four fixtures are evaluated: silence; a 150 Hz sinusoid of amplitude 0.5; Gaussian noise with standard deviation 0.1 and seed 41820; and the same sinusoid plus a 300 Hz harmonic of amplitude 0.2. Each contains 400 samples at the stated rate. The pole fit receives the original Hamming window, and the resulting coefficient finiteness, maximum pole radius, voicing, energy, and pitch are recorded. These are implementation checks, not speech intelligibility or speaker generalization tests.

The related benchmark uses order eighteen, a nominal 64-entry spectral codebook, regularized autocorrelation fitting, and source-filter reconstruction. Inspection follows its codebook construction and evaluation dataflow. No private voice recording is copied into this publication package, and no listening claims are inferred from the presence of audio files. The study does not assign performance to unavailable or unqualified speech results.

7 Results

Table 1: Deterministic synthetic fixtures. Pitch zero denotes an unvoiced decision.

Fixture	Voiced	Pitch (Hz)	Largest pole radius
Silence	No	0	0
150 Hz sine	Yes	150.9434	0.9499991
Gaussian noise	No	0	0.8514393
150 + 300 Hz	Yes	150.9434	0.9498142

All four filters have finite coefficients and pole radii below one. The pitch estimate is quantized by the integer lag search: a lag of 106 samples gives approximately 150.94 Hz. This validates a simple signal pathway, not robustness to pitch doubling, transients, background noise, or diverse speakers. No superiority over standard LPC pitch and envelope estimators follows.

The three-second demonstration processes 119 complete frames because its loop excludes the final valid 400-sample frame. Its printed rate divides an assumed per-frame bit count by the original duration, not the actual serialized payload or reconstructed duration. No bitstream is written and no independent decoder reloads it. These observations invalidate the nominal low-rate claim without requiring a subjective judgment about sound quality.

In the automatic benchmark, spectral descriptors from the recording are clustered and converted to medoid filter choices; that same recording is then reconstructed and scored. The codebook is not an independently trained shared object evaluated on held-out speech. Pitch and energy are also passed directly without quantized transmission. A six-bit index for 64 choices would not account for the codebook, pitch, gain, voicing, synchronization, or error protection.

Its log-spectral-distance score uses pre-emphasized active frames, a Hamming window, and a fixed spectral floor. Threshold labels describing transparent or intelligible communication are not supported by a listening study in the retained evidence. Magnitude-spectrum agreement is not interchangeable with linguistic intelligibility, perceptual quality, or task performance. The source audit therefore reports the mechanism and qualification failures, rather than a codec leaderboard number.

8 Discussion

The operator viewpoint remains useful for separating excitation, resonances, and synthesis state. It can help an engineer reason about stability and which information must cross a communication boundary. In this implementation those ingredients largely instantiate classical signal processing. A scientific contribution beyond this diagnostic report would require a distinctive representation, quantization or estimation method with a validated benefit; terminology alone cannot supply it.

The supplement contains the unmodified analysis sources, fingerprints, complete fixture definitions, numerical outcomes, and explicit payload arithmetic. No microphone is accessed and no speech is played. The tests can be repeated without the original personal recording. The resulting evidence supports finite stable analysis on these fixtures and exposes missing codec semantics; it does not establish a deployable low-bandwidth communication system.

9 Application boundary and research implication

The operator viewpoint is most useful here as a disciplined model of what must be encoded and what can be regenerated. A future learned residual or quantizer could be a genuine ML contribution, but the current evidence is a classical parametric prototype and a qualification audit. The revision does not convert assumed bitrates into measurements.

10 Conclusion

A pole-based synthesis model is not yet a compressed speech format. The audited prototype demonstrates classical frame analysis and source-filter reconstruction, but lacks a complete serialized decoder interface and qualified rate-quality evidence. Explicit payload accounting and deterministic stability checks identify the gap while preserving the useful implementation. The appropriate current output is a transparent technical study, not a claim of spectacular speech compression.

References

- [1] J. Makhoul. Linear Prediction: A Tutorial Review. *Proceedings of the IEEE* 63(4), 561-580, 1975.
<https://doi.org/10.1109/PROC.1975.9792>
- [2] N. Zeghidour et al. SoundStream: An End-to-End Neural Audio Codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2022. <https://arxiv.org/abs/2107.03312>