Granulated Low-bitrate Mixture

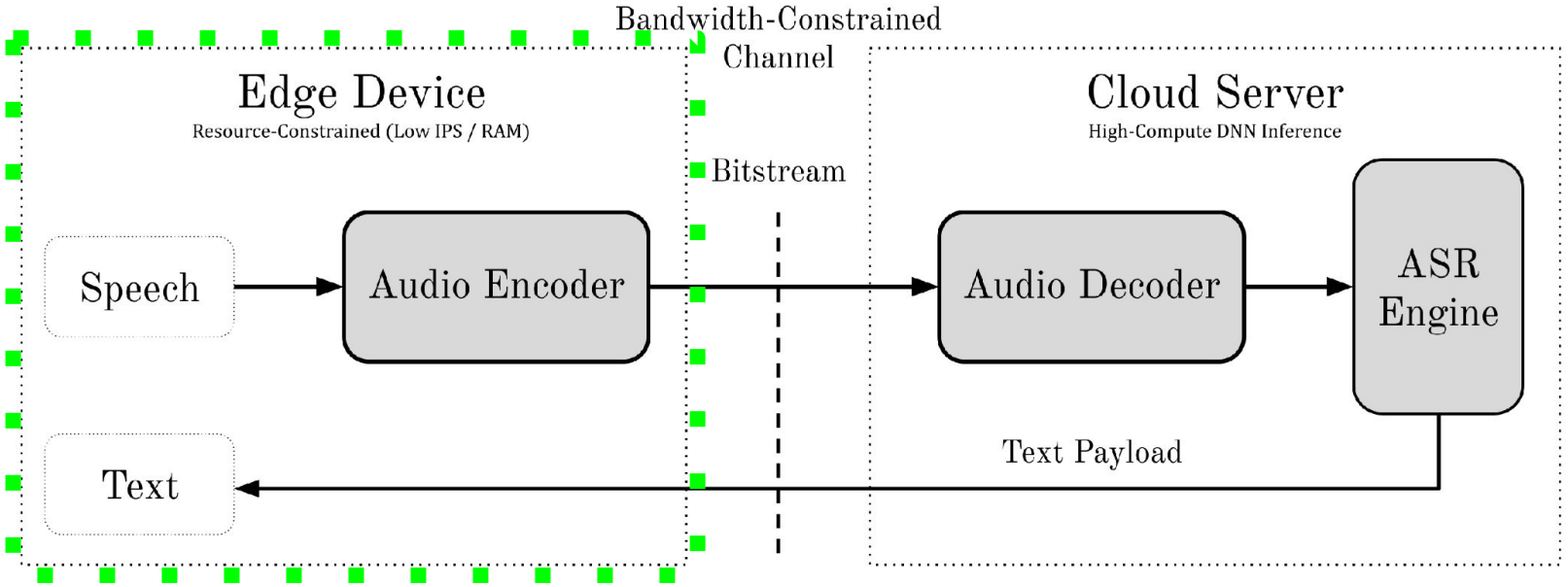
An Open-Source Lightweight ASR Codec

Ellison Murray · Alexander Chen · Anya Knudsen · Adam Jean

Morriel Kasher · Predrag Spasojevic

3.01% WER · 24 kbps · 168 bytes RAM · *zero* floating-point ops

Edge devices cannot handle speech recognition



Left: small encoder on the device. Right: heavy model in the cloud, where compute doesn't matter.

- Speech recognition models are huge.
- Earbuds and smart glasses aren't big enough → send the audio somewhere else.

Sending raw audio **drains** the battery and the bandwidth → compress the audio first, without using a lot of resources, and then send it.

THE GOAL

What we were optimizing for

Instructions

How much math the encoder does per second of audio.

Less instructions = more battery life

Memory

How much RAM the encoder needs to hold while it works.

Does it even fit in a small device?

Bitrate

How many bits per second we send over the link.

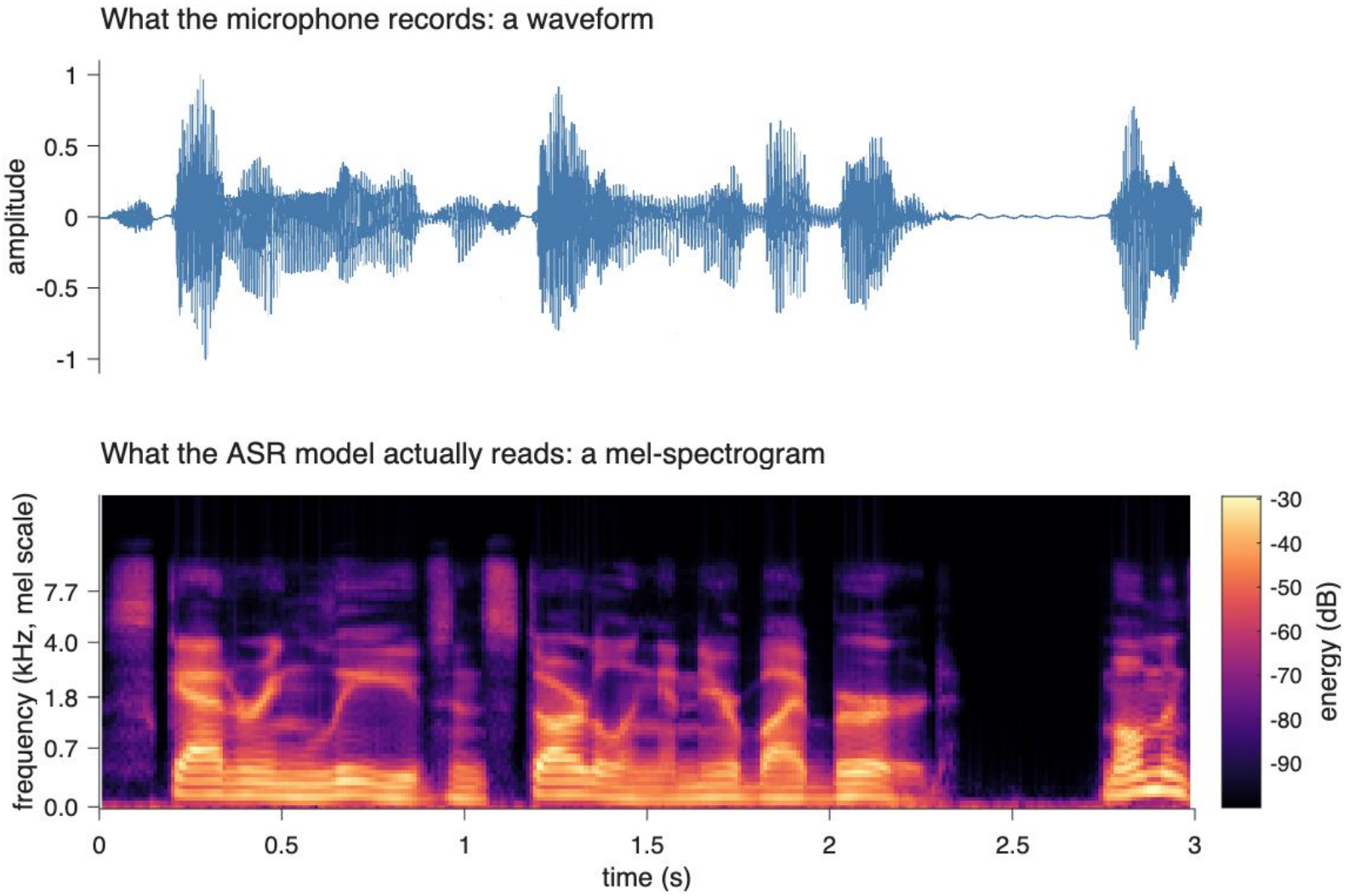
Bandwidth cost

We are NOT optimizing for how good it sounds.

Codecs like MP3 and Opus are tuned so audio sounds good to a human ear, but no human ever hears our audio.

CONCEPT 1 OF 2

ASR model



Mel-spectrogram

A picture of sound:

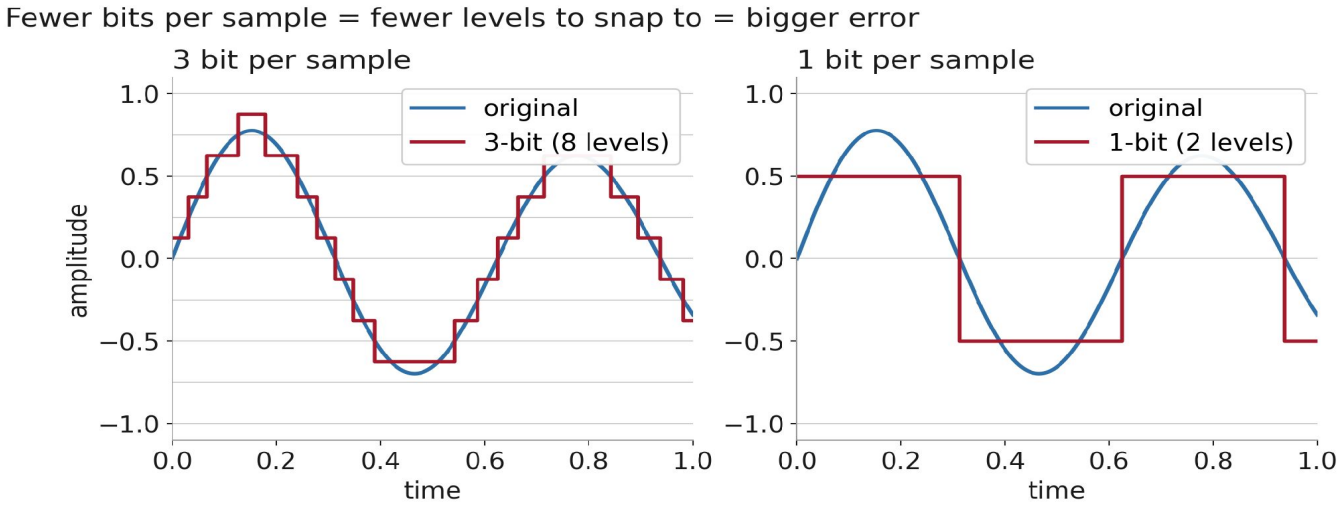
- y-axis: pitch / frequency
- brightness intensity is energy

Speech recognition convert the waveform to this picture (a spectrogram) to “read” it.

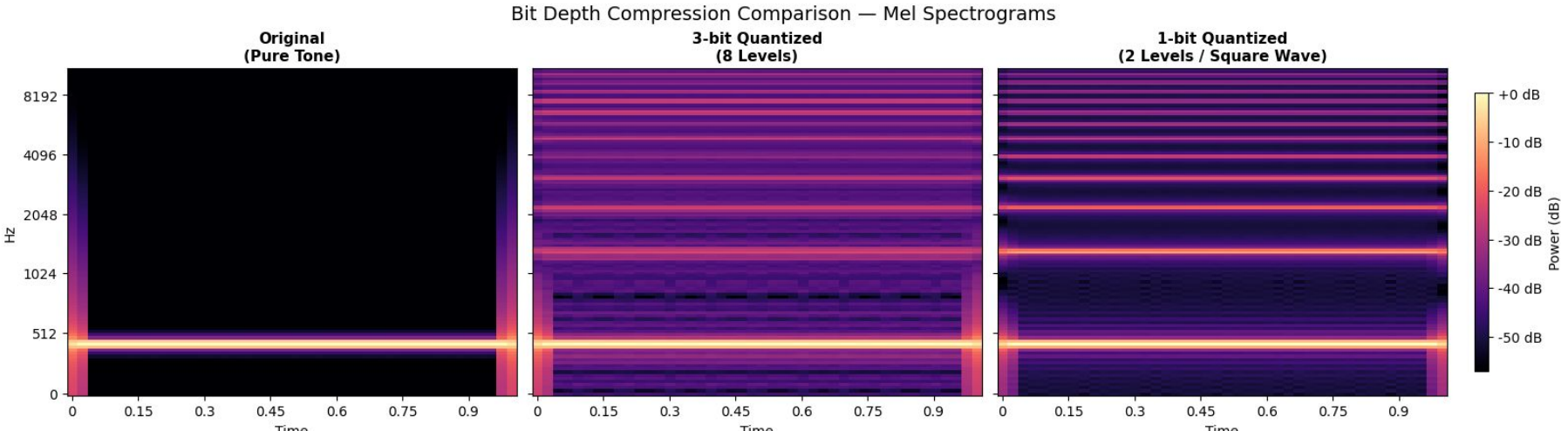
"Mel": the pitch / frequency axis is spaced the way human hearing is:

- stretched at lower frequencies and squeezed at higher

Low bitrate Quantization



Quantization: every sample gets rounded to the nearest available level.



- Normal audio uses 16 bits per sample: 65,536 levels
- We use 1, 2 or 3 bits: 2, 4 or 8 levels
- Savings in bitrate
- Damage from rounding error
- Lose information
- Harmonic distortion

CONCEPT 2 OF 2

Subtractive dithering

1 Add

The encoder adds a small random number v (noise) to the sample.

2 Round

Quantize as usual. Rounding decision is randomized.

3 Subtract

The decoder subtracts the exact same v back off.

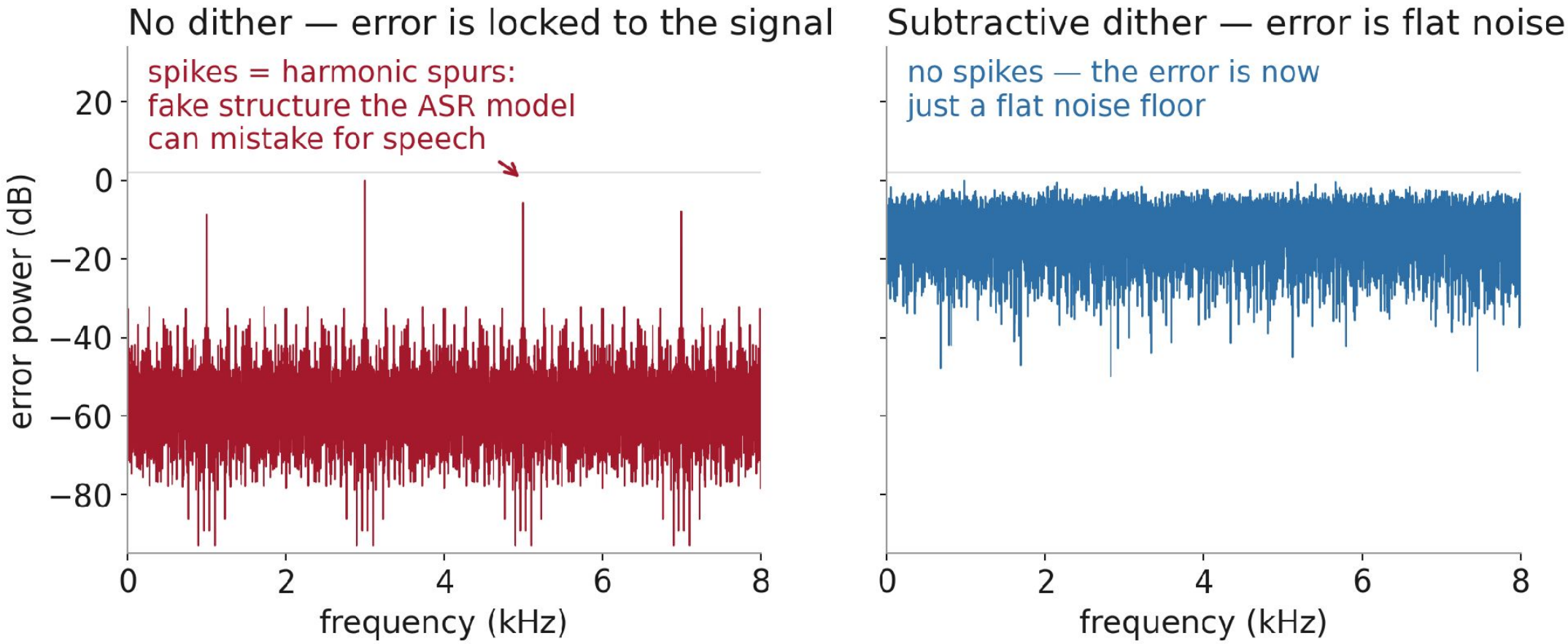
Why bother? Randomizing the rounding

The error no longer repeats with the waveform, so it stops creating false frequency and just becomes random background noise. Automatic Speech Recognition models handle that type of noise better than structured errors that can look like real signal features or actual syllables.

CONCEPT 2 OF 2

Dithering results, visualized

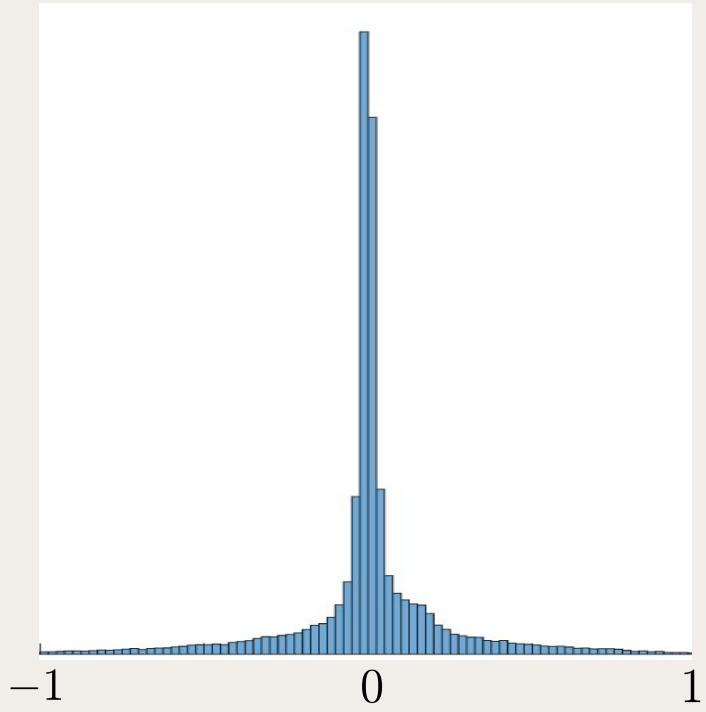
Same 3-bit quantizer, same signal — the only change is the dither

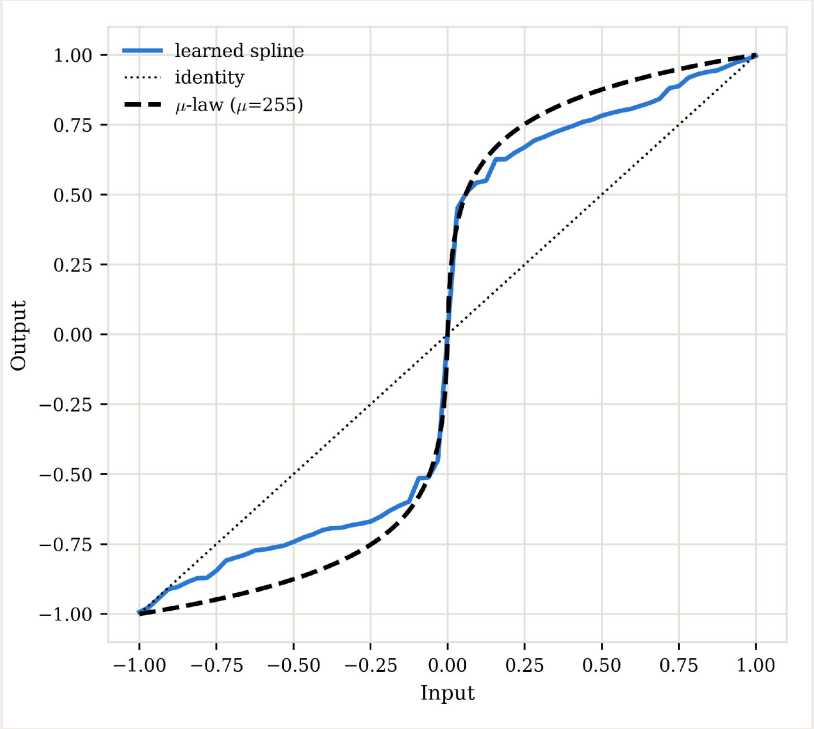


WHAT WE BUILT

Fine-tuning and training the ASR

- Speech spends most of its time near zero (doesn't get that loud), and the louder levels almost never get used.
- We trained a curve to reshape the signal before rounding, so all the levels get used more equally
- It keeps as much information about the original as possible





Blue = the learned curve. Dashed = μ-law.

It rediscovered μ-law companding: the curve telephone networks have used for decades.

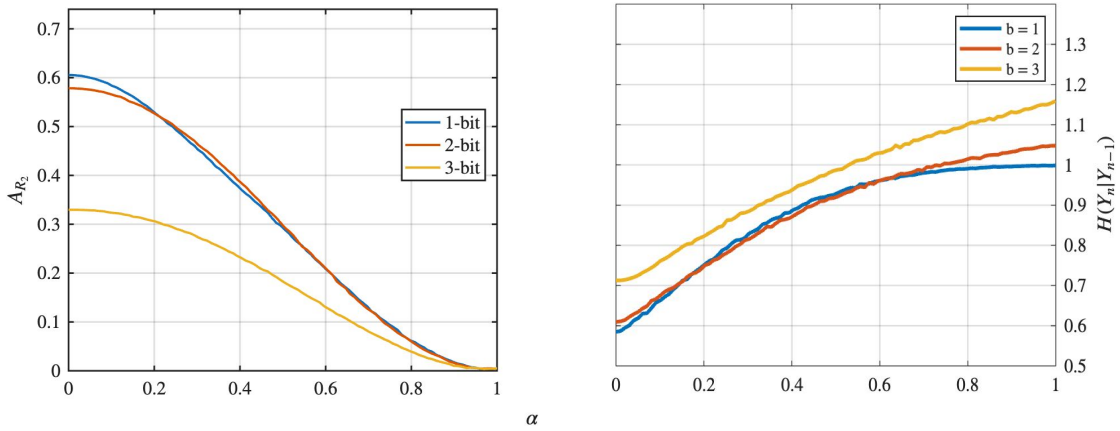
WHAT WE BUILT

Dithering α and noise level

α = 0 → no dither α = 1 → full dither in between → partial

Turning α up

- Cleans up the fake structure
- But costs information
- And costs bitrate: samples get harder to compress

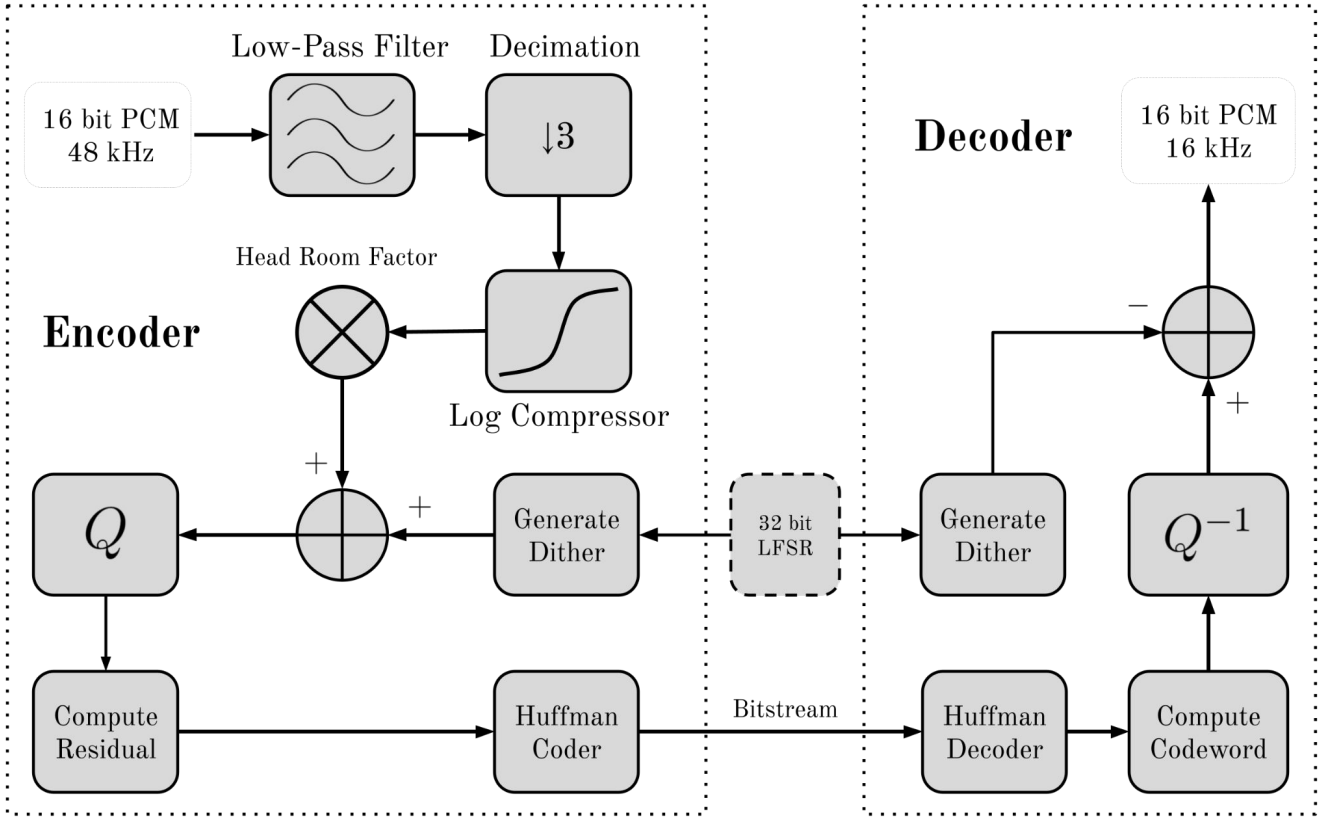


Left: fake structure falls as α rises. Right: bits needed rises as α rises.

We swept α and picked the best spot for each bit depth.

Dithering helped most at 2 bits (23% cut in errors) but created more errors at 1 bit.

Full encoder and decoder



Written in C

- .glx file format, has its own header and error check
- Written entirely in integer arithmetic + zero floating-point operations.
  - These chips have no floating-point unit.

Tested on a simulated RISC-V chip: helps estimate the compute capabilities of edge devices.

RESULTS

Results: far cheaper, slightly less accurate

Codec	Encoder work (MIPS)	Bitrate (kbps)
G.711	0.345	64.0
GLX	2.25	23.7
G.726	9.66	24.0
MP3	16.9	24.0
SILK	24.2	23.4

4.16×

less encoder work than any codec at a similar bitrate

3.01%

word error rate on clean speech **5–10% considered production-grade**

MIPS: Millions of instructions per second of audio (less is better for battery life)

At the same bitrate our accuracy is **a bit worse** than MP3 or SILK. When using a chip with no floating-point unit and a battery to protect, this trade is worth it.

A summary:

- Showed you can compress speech for a machine listener, not a human one — and that the rules about compression change when you do
- Made "how much noise to add" into a tunable dial, and measured what it costs you
- A real, open-source, integer-only codec — and benchmarked it on the hardware it is meant for

Thanks for listening!

Questions?

Thanks for listening!

Questions?

Memory Footprint

Stage	ROM (bytes)	RAM (bytes)
CRC (header check)	248	0
Resampling	450	80
Compression	370	0
Dithering	244	4
Quantization	90	2
Coding	385	36
Total		**1171**

METHOD

Mixture dither with one dial, α

$$V \sim f_V(v) = \alpha \cdot \Pi_{\alpha\Delta}(v) + (1 - \alpha) \cdot \delta(v), \quad \alpha \in [0, 1]$$

- Subtractive dithering makes the quantization error independent of the input, flattening its spectrum — which matters because ASR front ends read mel-spectrograms, so harmonic spurs hurt more than raw error power.
- α mixes uniform dither with a point mass at zero, continuously interpolating between none and full.
- Schuchman: full decorrelation needs Φ_V to vanish at multiples of $2\pi/\Delta$ — and $|\Phi_V| \to 0$ as $\alpha \to 1$.
- AR_2, the ℓ_2 norm of the error autocorrelation, falls monotonically in α and b: predictable partial whitening.

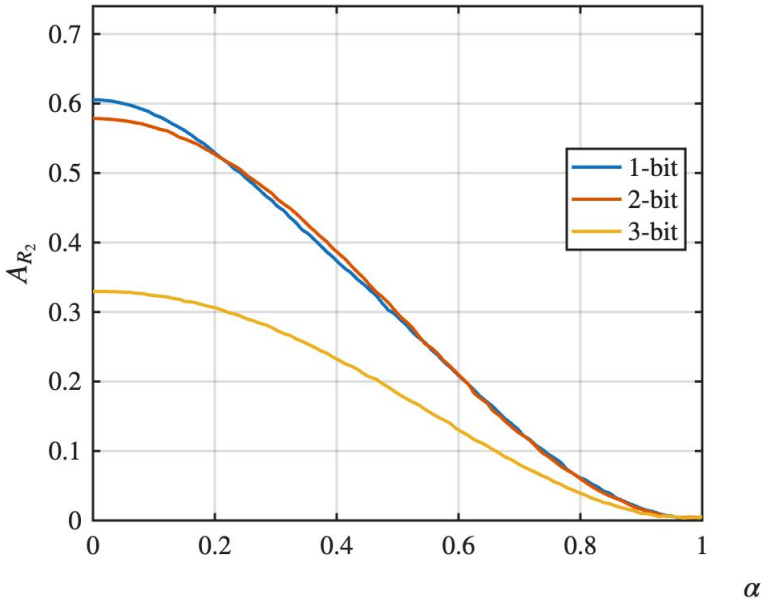


Fig. 5a — A_{R_2} of ε_S vs. α (max lag 5).

ANALYSIS

Measured α trade-off

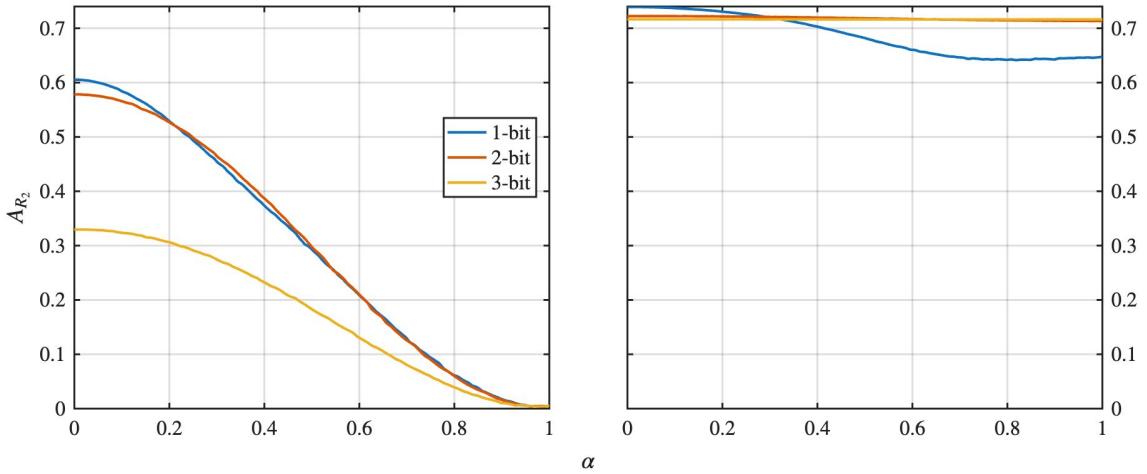


Fig. 5 — AR_2 (error correlation) vs. α.

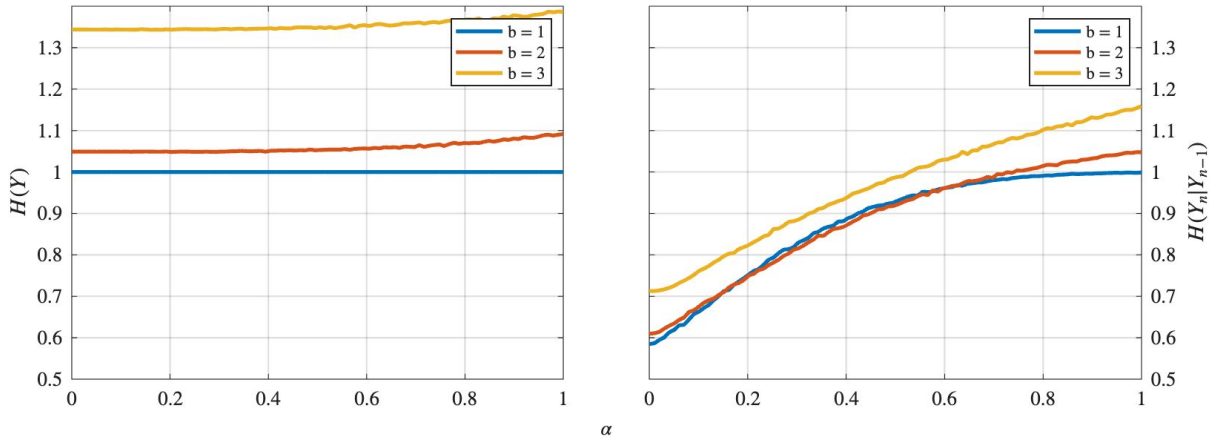


Fig. 6 — (a) H(Y), (b) H(Y□|Y□₋₁) vs. α.

- AR_2 falls monotonically in α and in bit depth: whitening is continuous and predictable.
- Marginal entropy H(Y) barely moves; the first-order conditional entropy climbs, which is what raises the effective bitrate.
- Preserved information I(X;Z) is provably monotonically decreasing in α (Sec. III of the paper).

Accuracy Results

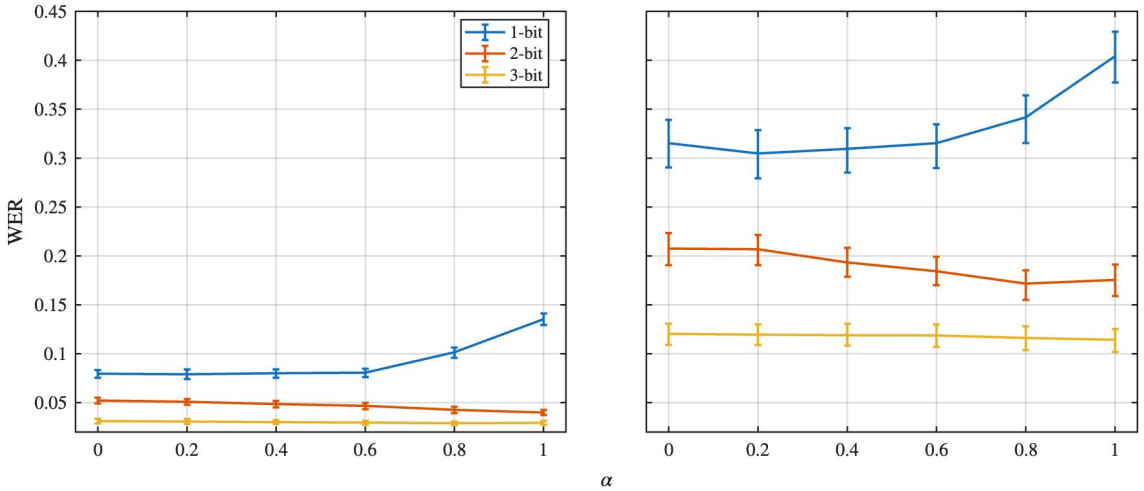


Fig. 9 — WER vs. α, (a) test-clean (b) test-other.

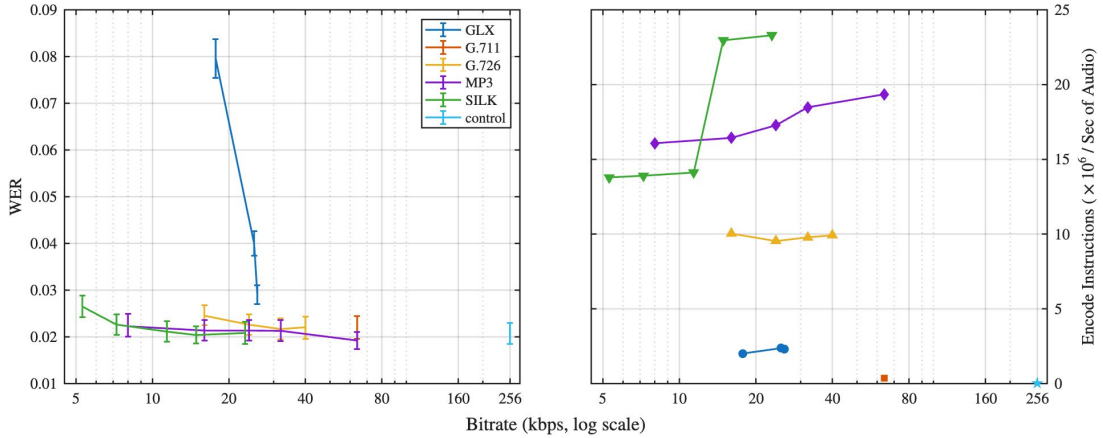


Fig. 10 — GLX vs. baselines: WER and encoder cost.

- 2,620 held-out LibriSpeech utterances, Whisper large-v3, 95% bootstrap confidence intervals.
- Going from test-clean to test-other costs GLX 2.88× WER on average, versus 2.14× for the baselines — low-resolution scalar quantization is more sensitive in noise.

METHOD

How accuracy was measured

- Peak-normalize, resample 16 → 48 kHz, run through the codec, restore to 16 kHz float and match original length
- Transcribe with Whisper large-v3
- Score after the Whisper text normalizer every codec **and** the uncoded control

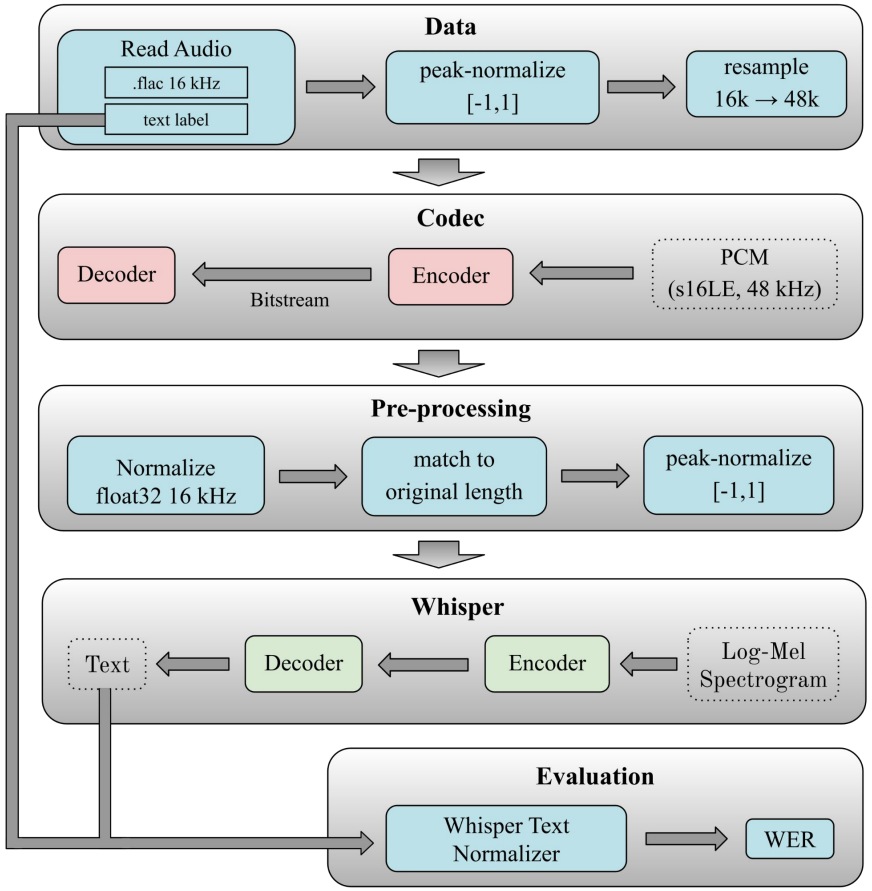


Fig. 8 — Evaluation pipeline.

2,620

held-out utterances from LibriSpeech test-clean and test-other.