DELEXWAV: AN AUDIBLE REPRESENTATION OF FINE-GRAINED PROSODY

Anonymous authors
Paper under double-blind review

ABSTRACT

Speech carries both lexical content and prosody. Explicit acoustic features describe selected aspects of prosody, while learned vectors and tokens are heard through a downstream generator. We introduce DelexWav, a learned waveform representation that preserves fine-grained prosody while suppressing recoverable lexical content. DelexWav provides utterance-specific prosodic conditioning for speech synthesis. Its waveform form enables direct listening, acoustic comparison with the original speech, and local editing. We construct fixed-phone targets using phoneme-level conditional-flow TTS and use them as the main supervision for a neural codec that extracts DelexWav directly from speech, without transcripts or phone alignments at inference. On 1,234 LibriSpeech test-clean utterances, DelexWav preserves pitch and energy variation with correlations of 0.96 and 0.85. A phone recognizer trained on DelexWav recovers no complete source words. The English-trained model also preserves prosodic variation in cross-lingual tests without retraining. Speech synthesis experiments show that DelexWav provides fine-grained prosodic conditioning, with local edits producing corresponding changes in the generated speech. Samples are provided in the supplementary material.

1 INTRODUCTION

Speech carries both lexical content and prosody. Here, fine-grained prosody refers to utterance-specific structure that varies at frame and phone scales. Pitch, energy, voicing, pauses, speaking rate, and segment duration provide measurable but incomplete views of this structure, whose full organization is difficult to reduce to explicit features.

Existing representations follow two main directions. Explicit features describe selected aspects of prosody but leave its remaining structure unspecified. Learned vectors or token streams may capture richer information, but their acoustic effects can only be heard through a downstream model. We therefore seek a learned waveform representation that preserves rich prosodic structure while suppressing recoverable lexical content. Such a representation can be listened to directly and compared with source speech using the same acoustic measurements, revealing which patterns are preserved or changed.

We introduce DELEXWAV, an audible waveform representation for fine-grained prosody disentanglement. Inspired by delexicalized and unintelligible speech synthesis (Sonntag & Portele, 1998; Ramus & Mehler, 1999), we construct targets with a single fixed phoneme to reduce phonetic variation. In our comparison, a codec trained on neural fixed-phone targets yields fewer recoverable source-text fragments than one trained on random-phone rule targets.

To construct these targets, we use a two-stage phoneme-level conditional-flow TTS pipeline. Both stages are trained to reconstruct original speech; the final model generates fixed-phone targets by replacing its input phone sequence at inference. With source-speaker conditioning, these targets better retain the original speaker's voice characteristics than fixed-phone rule synthesis. We use them as the main supervision for a neural codec that extracts DELEXWAV directly from speech, without transcripts or phone alignments at inference. The decoded waveform, rather than the hidden quantized code, is DELEXWAV.

We first evaluate what DELEXWAV preserves and suppresses. Direct comparisons with the original speech measure prosody preservation, while trained phone recognizers test lexical recovery.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

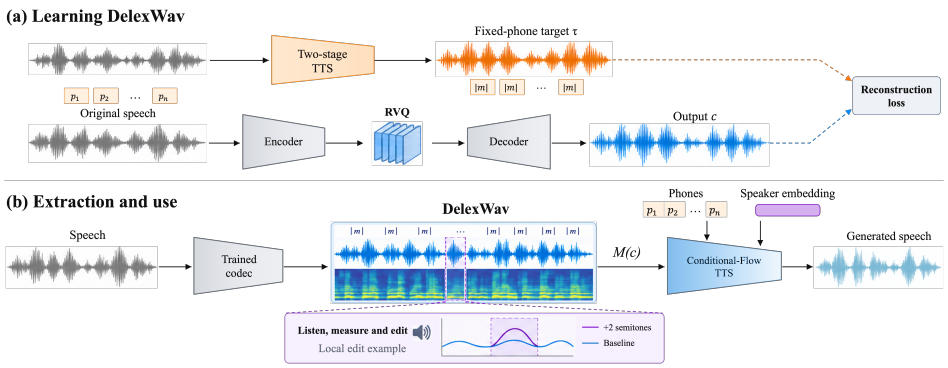


Figure 1: Construction and use of DELEXWAV. (a) Two-stage TTS generates fixed-phone targets τ to supervise a codec that maps original speech to the waveform c. (b) The trained codec extracts DELEXWAV for listening, measurement, and local editing. Its mel spectrogram $M(c)$, phones, and a speaker embedding condition a speech generator.

Cross-lingual tests further assess the extractor's generalization without retraining. We then evaluate DELEXWAV as a prosody condition for a separately trained conditional-flow TTS model. Same-text reconstruction tests prosody preservation in generated speech, while phone switching and local edits test control over pronunciation and prosody.

On 1,234 utterances from LibriSpeech test-clean, direct measurements show that DELEXWAV retains utterance-specific pitch and energy trajectories. The tested phone recognizer succeeds on original speech but recovers no complete source words from DELEXWAV. As a reconstruction condition, DELEXWAV yields more accurate phone timing than the explicit F0+Energy features, but reproduces pitch and energy less accurately. Compared with FACodecs learned prosody codes, DELEXWAV improves phone timing, pitch, and energy reconstruction, but has lower voicing agreement and F0 recovery. Switching the phone sequence changes the generated pronunciation, while local edits enable control of pitch and timing, sometimes at the cost of content accuracy. The extractor also preserves measurable prosodic variation in cross-lingual tests. Our contributions are:

- DELEXWAV, a waveform representation of rich, utterance-specific fine-grained prosody that can be listened to, measured, and edited directly;
- A construction method that combines conditional-flow fixed-phone targets with neural codec training to extract DELEXWAV from speech without transcripts or phone alignments;
- Experimental evidence that DELEXWAV preserves fine-grained prosody while strongly suppressing recoverable lexical content, supporting prosody disentanglement. We further demonstrate cross-lingual transfer of the extractor and applications in speech reconstruction and control.

2 RELATED WORK

Fine-grained prosody representations. Phone- and word-level latent variables model interpretable prosody at multiple resolutions (Sun et al., 2020). SpeechSplit separates content, timbre, pitch, and rhythm through information bottlenecks (Qian et al., 2020), while NANSY uses perturbed analysis features for controllable reconstruction (Choi et al., 2021). Explicit acoustic representations further support direct edits to pitch, loudness, duration, pronunciation, and timbre (Morrison et al., 2024). These approaches represent prosody through latent variables or selected acoustic features.

Prosody codecs. NaturalSpeech 3 factorizes a codec into content, prosody, timbre, and acoustic-detail streams (Ju et al., 2024). Phoneme-level RVQ uses linguistic and speaker conditioning to learn discrete prosody representations (Karapiperis et al., 2024). ProsoCodec instead models prosody as the variation remaining after conditioning on text and speaker identity (Choi et al., 2026). These methods represent prosody as codes for speech generation. In DELEXWAV, the decoded fixed-phone waveform is itself the prosody representation.

Audible delexicalized speech. PURR reduces segmental content to support perceptual evaluation of prosody (Sonntag & Portele, 1998), while speech resynthesis has also been used to study rhythm and intonation in language discrimination (Ramus & Mehler, 1999). Building on these resynthesis methods, we construct fixed-phone targets with conditional-flow TTS and train a codec to extract DELEXWAV directly from speech, without transcripts or phone alignments at inference. Prosodic features also share information with text (Wolf et al., 2023), motivating direct tests of lexical recovery from the extracted waveform.

3 PROBLEM SETUP

Let x be a speech waveform, t its transcript, $P(t)$ the corresponding phone sequence, and s an external speaker embedding. A codec encoder E and quantizer Q encode x as $z_q = Q(E(x))$. The codec decoder converts z_q into an audible waveform c. Let $M(\cdot)$ denote the log-mel transform. A speech generator G predicts the source log-mel spectrogram from $M(c)$, $P(t)$, and s, and a vocoder V maps it to waveform,

$$\hat{x} = V(G(M(c), P(t), s)) . \tag{1}$$

We study the decoded waveform c as DELEXWAV. We evaluate prosody preservation and lexical suppression, and test fixed-phone switching and local prosody control.

Prosody preservation. We compare c directly with x to measure F0, energy, and voicing preservation. We also compare the reconstructed speech $V(G(M(c), P(t), s))$ with x to assess prosody preservation in downstream synthesis, including phone timing. Word accuracy is reported as a secondary reconstruction measure.

Lexical suppression. We train a recognizer A to recover source phonemes from c. Its recognition performance measures lexical recoverability under the tested model.

Fixed-phone control. With c fixed, we change the input phone sequence and examine the generated pronunciation.

Local control. Let g_I edit the pitch or duration of c within interval I. Keeping the phone sequence and speaker embedding fixed, we compare $V(G(M(g_I(c)), P(t), s))$ with the unedited reconstruction to assess the direction and temporal extent of the change.

4 METHOD

As shown in Figure 1, we construct fixed-phone targets from a rule-based waveform through two conditional-flow TTS stages. Both stages are trained to reconstruct original speech, with phone substitution applied at inference. These targets provide the main supervision for a codec that extracts DELEXWAV from speech. A separately trained speech generator then uses DELEXWAV for downstream reconstruction and control.

4.1 RULE TEMPLATE AND FIRST-STAGE TTS

A rule-based phoneme resynthesizer constructs the initial waveform d. With $\mathcal{I}(x)$ denoting the aligned source phone intervals and their F0 measurements, we write

$$p^{(0)} = \rho_0(P(t)), \qquad d = G_{\text{rule}}(\mathcal{I}(x), p^{(0)}). \tag{2}$$

Here ρ_0 maps the source phones to a small set of phonemes. The resulting waveform follows the source phone timing and F0 but has limited acoustic fidelity.

We train a first-stage conditional-flow TTS model (Chen et al., 2025) with $M(d)$, $P(t)$, and s as conditions and $M(x)$ as the target. After training, we replace the true phone sequence with $p^{(0)}$ to generate d_1. The mel spectrogram $M(d_1)$ provides the acoustic condition for the second stage.

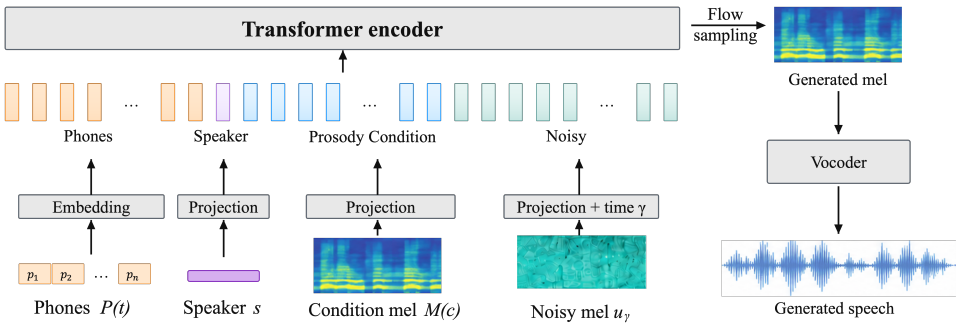


Figure 2: Conditional-flow TTS structure.

4.2 Second-stage TTS and fixed-phone target generation

Second-stage training. Define $m_{d_1} = M(d_1)$ and $m_x = M(x)$. We initialize a second TTS model from the first-stage checkpoint and fine-tune it with m_{d_1}, $P(t)$, and s as conditions and m_x as the reconstruction target. For $\epsilon \sim \mathcal{N}(0, I)$ and flow time $\gamma \sim \mathcal{U}(0, 1)$, we use the conditional flow-matching objective (Lipman et al., 2022):

$$\begin{aligned} u_\gamma &= (1-\gamma)\epsilon + \gamma m_x, & v^\star &= m_x - \epsilon, \\ \mathcal{L}_{\text{CFM}} &= \mathbb{E}\left[\|v_\theta(u_\gamma, \gamma \mid m_{d_1}, P(t), s) - v^\star\|_2^2\right]. \end{aligned} \tag{3}$$

Codec target generation. After fine-tuning, we replace the true phone sequence with $\rho_\star(P(t))$ to generate the codec targets. The operator $\rho_\star$ converts $P(t)$ to a sequence of the fixed phoneme $p^\star$. The acoustic condition m_{d_1} and speaker embedding s remain unchanged:

$$\widetilde{m} = \text{Flow}_\theta(m_{d_1}, \rho_\star(P(t)), s), \qquad \tau = V(\widetilde{m}). \tag{4}$$

Here Flow_θ denotes sampling with the learned velocity field. We generate τ offline and use it as the main target for codec training. Appendix A provides the rule-template and two-stage TTS implementation details.

4.3 Codec-based prosody extraction

A DAC-style neural codec (Kumar et al., 2023) learns to map source speech x to the DelexWav waveform c through an encoder, residual vector quantizer, and decoder:

$$\begin{aligned} z &= E(x), & r_0 &= z, \\ q_k &= Q_k(r_{k-1}), & r_k &= r_{k-1} - q_k, \\ z_q &= \sum_{k=1}^{K} q_k, & c &= D(z_q). \end{aligned} \tag{5}$$

Here K is the number of quantization stages, and r_k is the residual after stage k. The generated targets τ may omit acoustic detail needed to represent the source prosody. We therefore replace τ_i with the source waveform x_i for a small, fixed subset of codec-training utterances:

$$y_i = \begin{cases} x_i, & i \in \mathcal{S}_\alpha, \\ \tau_i, & i \notin \mathcal{S}_\alpha, \end{cases} \tag{6}$$

Here $\mathcal{S}_\alpha$ is a fixed subset with nominal fraction α, and the codec input is always x_i. Because source targets also contain the original phones, we use them for only a small fraction of utterances and evaluate lexical recovery from the resulting representation. We optimize the codec using reconstruction, feature-matching, and adversarial losses against the selected target, together with RVQ commitment and codebook losses. After training, the codec extracts DelexWav directly from the source waveform.

4.4 DOWNSTREAM SPEECH GENERATOR

To use DELEXWAV for speech reconstruction, we train a separate F5-TTS-style speech generator G_ω (Chen et al., 2025) from scratch, using $M(c)$, $P(t)$, and s as conditions and $M(x)$ as the reconstruction target:

$$\hat{x} = V(G_\omega(M(c), P(t), s)) . \tag{7}$$

The codec remains frozen during speech-generator training, and G_ω does not share parameters with the teacher used to generate τ. The generator architecture is shown in Figure 2.

5 EXPERIMENTAL SETUP

5.1 DATA AND COMMON SETUP

Codec and speech-generator training use the LibriTTS-R train and dev splits (Koizumi et al., 2023); speech is resampled to 16 kHz and peak-normalized. The CTC recognizers are trained on Libri-Heavy (Kang et al., 2024). Unless specified otherwise, experiments use the final 350k DELEXWAV codec with fixed phone $p^\star = \text{|m|}$ and original-speech target fraction $\alpha = 0.1$.

The main test set contains 1,234 LibriSpeech test-clean utterances (Panayotov et al., 2015) from 40 speakers, with durations of 4–10 s, following the selection used by MELLE (Meng et al., 2025)[1]. Phone switching and local editing use a fixed 200-utterance subset, selecting one utterance from each of five duration quintiles per speaker. Cross-lingual extraction uses 300 Emilia (He et al., 2024) utterances, 100 each in English, Mandarin, and German.

5.2 DIRECT REPRESENTATION EVALUATION

We evaluate prosody preservation by comparing each waveform with the corresponding original speech x. The baselines are the initial MBROLA (Dutoit et al., 1996) |sHi| template and an all-|m| MBROLA waveform. The latter provides a rule-based comparison with the same fixed phone. Vocoder reconstruction $x_{\text{voc}} = V(M(x))$ provides a reference for the acoustic measurements.

Lexical recovery is tested with independently trained Conformer–CTC recognizers (Gulati et al., 2020; Graves et al., 2006) for original-speech mel, DELEXWAV mel, and the two-channel F0+Energy sequence. All use the same 121-symbol eSpeak vocabulary. Each recognizer receives only its corresponding features. Original speech is the positive control.

For cross-lingual evaluation, the same English-trained codec processes the Emilia speech without retraining, transcripts, or phone alignments. Acoustic measurements follow the direct evaluation protocol, with Emilia English providing a same-corpus reference.

5.3 DOWNSTREAM RECONSTRUCTION AND CONTROL

We train separate conditional-flow speech generators conditioned on DELEXWAV mel, F0+Energy, or FACodec prosody vectors (Ju et al., 2024). The three systems use the same transformer architecture, training data, and optimization schedule for 400k updates, with representation-specific input front ends. All receive the same text and speaker conditions and use 32 flow steps with the same frozen BigVGAN-v2 vocoder (Lee et al., 2022).

Phone switching holds DELEXWAV and the speaker condition fixed while replacing the phone sequence with all-|m| or all-|s|. We measure the change in pronunciation and prosody.

Local editing keeps the phone input fixed and modifies only the middle 35–65% of DELEXWAV using pitch shifts of ± 2 semitones or time scales of 1.25 and 0.80. Time-stretch controls use F0+Energy editing and conventional phase-vocoder stretching of the unedited DELEXWAV-generated speech. Each system is compared with its own unedited output; time-edited generation follows the edited input length.

[1] https://github.com/Shy-98/MELLE

5.4 EVALUATION METRICS

Prosody preservation. Reference F0 frames are voiced in both x and x_{voc}. F0 recovery is the fraction with detectable output F0; F0 MAE (cents) and log-F0 correlation measure accuracy on the recovered frames. We also report log-energy correlation and voiced/unvoiced (VUV) agreement. Wavelet Prosody Toolkit (WPT) (Suni et al., 2017) correlations compare combined pitch, energy, and speaking-rate contours. For generated speech, phone-duration and within-word boundary errors (ms) measure timing, and AutoPCP (Barrault et al., 2023) measures prosodic similarity.

Lexical recovery. For CTC probes, we report phone error rate (PER), the number of error-free utterances, and the number of reference words recovered without phone errors. For reconstructed speech, we report Whisper (Radford et al., 2023) word error rate (WER). For codecs trained with different targets, we count outputs whose Whisper transcriptions contain at least three or five consecutive words from the original transcript.

Speaker similarity. SIM is the cosine similarity between CampPlus (Wang et al., 2023) embeddings of each waveform and its corresponding original speech. We also report speaker identification accuracy among the 40 speakers.

Phone switching and local control. Phone switching uses decoded |m| and |s| shares and PER against the original phone sequence, with voiced fraction and high-frequency energy as supporting measures. Local control measures pitch and duration changes inside and outside the edited interval, as well as phone-timing errors and changes in WER. After time stretching, DTW aligns edited and unedited speech to compare their pitch and energy contours. Appendix B gives the computation and aggregation details.

6 RESULTS

We first examine prosody preservation, lexical suppression, and cross-lingual extraction in DELEXWAV itself. We then evaluate speech reconstruction and control. Appendix D analyzes target construction and further codec training.

6.1 DIRECT PROSODY PRESERVATION

DELEXWAV retains source pitch and energy variation. As shown in Table 1, F0 correlation is 0.96, energy correlation is 0.85, and 90.96% of reference-voiced frames are recovered. The all-|m| rule has lower pitch error, whereas DELEXWAV better preserves the energy contour and voicing pattern.

Across all four time scales, DELEXWAV has higher WPT combined-contour correlations than both MBROLA waveforms, rising from 0.62 at 0.125 s to 0.88 at 1 s. Appendix C.1 separates the pitch and energy components. Together, these comparisons characterize the prosody retained in a waveform that can also be listened to directly. Figure 3 shows this direct comparison for one utterance.

Table 1: Direct prosody preservation on LibriSpeech test-clean. Frame-level metrics and WPT combined-contour correlations use the same waveform conditions.

Waveform	Frame-level prosody					Multi-scale WPT correlation			
	F0 rec.↑	F0 MAE (cents)↓	F0 corr.↑	Energy corr.↑	VUV↑	0.125 s↑	0.25 s↑	0.50 s↑	1.00 s↑
MBROLA |sHi|	62.83%	55.21	0.95	0.70	0.67	0.53	0.63	0.71	0.82
MBROLA |m|	92.88%	37.20	0.97	0.67	0.69	0.32	0.44	0.66	0.82
DELEXWAV	90.96%	51.96	0.96	0.85	0.71	0.62	0.73	0.81	0.88

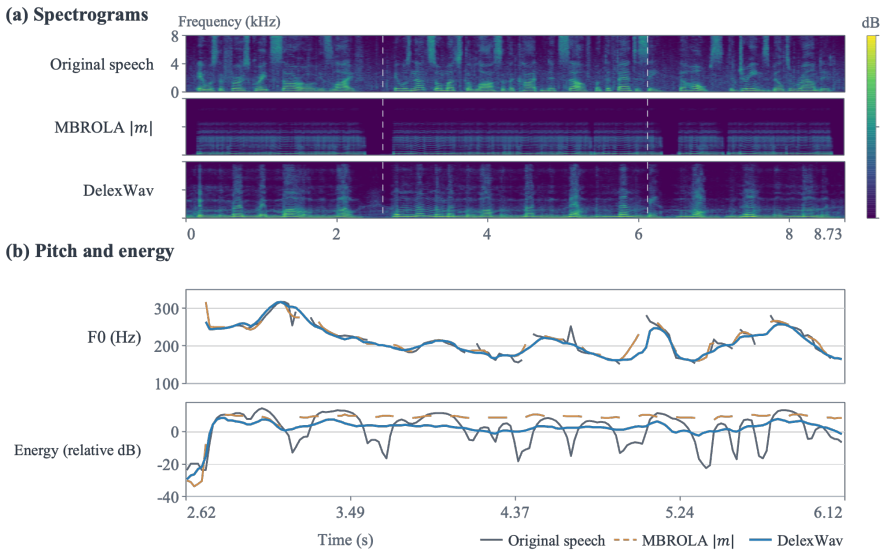


Figure 3: An example from LibriSpeech test-clean (utterance `1995-1837-0001`). (a) Spectrograms of original speech, MBROLA all-|m|, and DELEXWAV on a shared dB scale. The dashed lines mark the interval shown in (b), which compares F0 and mean-centered log-RMS energy.

6.2 LEXICAL SUPPRESSION

As shown in Table 2, the original-speech recognizer achieves 3.27% PER and recovers the complete phone sequences of 19,420 reference words, whereas the recognizers trained on DELEXWAV and F0+Energy recover none. These results show strong suppression under the tested recognizers.

Table 2: Direct lexical probing with independently trained recognizers and greedy CTC decoding.

Input	PER↓	Exact utterances↑	Recovered words↑
Original speech	3.27%	301	19,420 / 21,284
F0+Energy	93.04%	0	0 / 21,284
DELEXWAV	92.65%	0	0 / 21,284

6.3 CROSS-LINGUAL EXTRACTION

The English-trained extractor also retains pitch and energy variation in Mandarin and German without retraining. As shown in Table 3, pitch correlation is approximately 0.95 across all three language sets, while F0 recovery and energy preservation vary. Appendix C.2 reports auxiliary speech-recognition results.

Table 3: Direct extraction on 300 Emilia utterances, 100 per language. F0 MAE is in cents.

Language	F0 rec.↑	F0 MAE↓	F0 corr.↑	Energy corr.↑	VUV↑
English	90.79%	56.87	0.95	0.81	0.78
Mandarin	87.49%	68.90	0.95	0.86	0.73
German	94.14%	50.82	0.95	0.80	0.76

6.4 DOWNSTREAM SPEECH RECONSTRUCTION

Speech reconstructed from DELEXWAV has more accurate phone timing than speech reconstructed from F0+Energy or FACodec. As shown in Table 4, phone-duration and boundary errors are 7.55 and 16.15 ms, compared with 8.97 and 26.32 ms for F0+Energy.

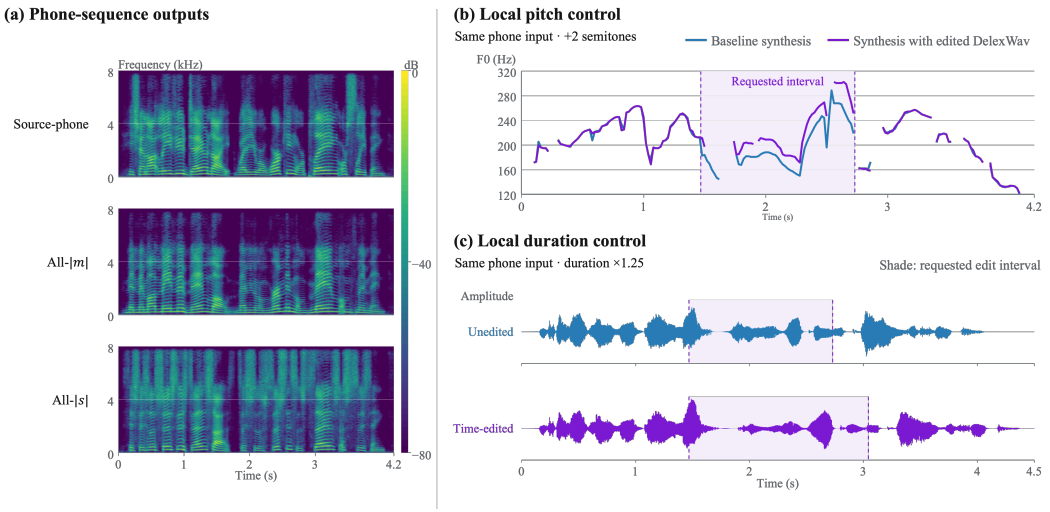


Figure 4: Phone switching and local prosody control for LibriSpeech test-clean utterance `5142-33396-0038`. All panels show generated speech. (a) Spectrograms with the original, all-|m|, and all-|s| phone inputs, with DELEXWAV held fixed. (b,c) F0 contours and waveforms before and after a local $+2$-semitone pitch shift or $\times 1.25$ time stretch of DELEXWAV, with the phone input fixed. Shading marks the requested edit intervals.

Table 4: Same-text reconstruction on 1,234 LibriSpeech test-clean utterances.

Output	Prosody					Phone timing		Output quality	
	VUV↑	Energy corr.↑	F0 corr.↑	F0 MAE (cents)↓	F0 rec.↑	Duration (ms)↓	Boundary (ms)↓	WER↓	SIM↑
Original speech	–	–	–	–	–	–	–	1.95%	–
x_{voc}	0.93	1.00	0.99	15.77	–	1.95	1.54	1.93%	0.99
F0+Energy	0.88	1.00	0.99	24.05	91.65%	8.97	26.32	35.52%	0.79
FACodec	0.83	0.89	0.94	83.10	87.31%	10.00	44.21	65.91%	0.76
DELEXWAV	0.81	0.95	0.97	46.42	79.12%	7.55	16.15	7.18%	0.77

F0+Energy more accurately preserves pitch, energy, and voicing. AutoPCP scores for speech generated from DELEXWAV and F0+Energy are close, at 3.25 and 3.24 (Appendix C.3). Synthesis also changes the DELEXWAV contours: F0 recovery falls from 90.96% in the representation to 79.12% in the generated speech, while energy correlation rises from 0.85 to 0.95.

As a secondary reconstruction measure, WER is 7.18% for DELEXWAV, compared with 35.52% for F0+Energy and 65.91% for FACodec. Original speech and vocoder reconstruction provide references at 1.95% and 1.93%. Together, these results show that DELEXWAV provides a useful prosody condition for speech reconstruction.

6.5 PHONE SWITCHING AND LOCAL CONTROL

DELEXWAV supports two forms of downstream control: changing the phone input changes pronunciation, while editing DELEXWAV changes local prosody.

Phone switching. With DELEXWAV held fixed, changing the phone input shifts the generated pronunciation. As shown in Table 5(a), the decoded |m| share rises from 1.76% to 10.90% under all-|m|, and the |s| share rises from 3.07% to 14.61% under all-|s|. Both outputs also depart further from the original phone sequence. The natural-speech recognizer produces mixed decodes of these fixed-phone outputs, so the shares indicate response direction rather than phone accuracy (Appendix C.4).

The pronunciation change retains much of the source prosody: all-ɪmɪ has an F0 correlation of 0.95, close to 0.96 with source phones, although pitch error increases. Energy correlation is 0.90 for both fixed-phone outputs, compared with 0.95 for source-phone outputs. As shown in Figure 4(a), switching the phone input changes the spectral structure of the generated speech. This phone-switching response does not yet extend to reliable sentence replacement (Appendix C.5).

Local editing. With the phone input fixed, editing DELEXWAV changes prosody in the corresponding output interval. Figure 4(b,c) shows the generated F0 contours and waveforms before and after editing DELEXWAV. As shown in Table 5(b), pitch edits of $+2$ and -2 semitones produce median inside shifts of $+190$ and -220 cents, with a median signed shift of 0 cents outside. CTC-aligned speech segments lengthen or shorten more within the edited interval than outside it.

Local editing increases WER, indicating reduced word accuracy after prosody modification. For local time stretching, F0+Energy and conventional waveform stretching preserve pitch and energy more accurately than editing DELEXWAV (Appendix C.6).

Table 5: Phone switching and local editing on 200 utterances. In (b), inside/outside effects are paired changes from the unedited output; duration ratios use summed CTC-symbol durations.

(a) Phone switching.

Phone input	Pronunciation response			Prosody preservation			
	ɪmɪ share	ɪsɪ share	PER	F0 corr.↑	F0 MAE (cents)↓	Pooled F0 rec.↑	Energy corr.↑
Source phones	1.76%	3.07%	9.05%	0.96	48.06	78.08%	0.95
ɪmɪ	10.90%	0.23%	70.48%	0.95	57.35	86.86%	0.90
ɪsɪ	0.11%	14.61%	71.77%	–	–	–	0.90

(b) Local editing.

Edit	Inside effect	Outside effect	Direction↑	WER↓
None	–	–	–	10.35%
Pitch $+2$ semitones	$+190$ cents	0 cents	97.21%	29.95%
Pitch -2 semitones	-220 cents	0 cents	98.37%	46.29%
Time $\times 1.25$	duration $\times 1.11$	$\times 1.01$	81.00%	20.61%
Time $\times 0.80$	duration $\times 0.85$	$\times 0.98$	97.00%	24.82%

7 LIMITATIONS

Lexical suppression is evaluated under the tested CTC recognizers; the DELEXWAV recognizer produces a constant hypothesis. The ɪmɪ-based teacher favors voicing and weakly preserves internal pauses. Local edits increase WER, and cross-text synthesis remains unreliable (Appendix C.5).

The 250k-to-350k comparison combines additional codec training with 10% original-speech targets, so it does not isolate the effect of those targets. Training uses English audiobook speech, and cross-lingual evaluation covers Mandarin and German only.

8 CONCLUSION

We presented DelexWav, a learned waveform representation for fine-grained prosody disentanglement. It retains time-varying prosodic structure while strongly suppressing complete source-word recovery under the tested recognizer. The extractor transfers across languages without retraining, and downstream synthesis demonstrates its use in reconstruction and control. By representing prosody in waveform form, DelexWav enables listening, acoustic comparison with the original speech, and local editing within the same representation.

AI USE STATEMENT

Generative AI tools assisted with research planning, literature search, implementation and result analysis, manuscript drafting, and figure design. Reported numerical results were computed from experimental outputs. The authors reviewed the AI-assisted work, verified the reported results, and take responsibility for the final content.

ETHICS STATEMENT

Experiments use publicly available speech corpora. Speaker-conditioned synthesis may be misused for impersonation. Any release should respect dataset licenses and speaker privacy and clearly identify generated speech.

REPRODUCIBILITY STATEMENT

Sections 4 and 5 describe the method and experimental setup. Appendices A and B provide training details and metric definitions, while Appendix C reports supplementary results.

REFERENCES

Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, Ning Dong, Mark Duppenthaler, Paul-Ambroise Duquenne, Brian Ellis, Hady Elsahar, Justin Haaheim, et al. Seamless: Multilingual expressive and streaming speech translation. *arXiv preprint arXiv:2312.05187*, 2023.

Mathieu Bernard and Hadrien Titeux. Phonemizer: Text to phones transcription for multiple languages in python. *Journal of Open Source Software*, 6(68):3958, 2021.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6255–6271, 2025.

Hyeong-Seok Choi, Juheon Lee, Wansoo Kim, Jie Lee, Hoon Heo, and Kyogu Lee. Neural analysis and synthesis: Reconstructing speech from self-supervised representations. *Advances in Neural Information Processing Systems*, 34:16251–16265, 2021.

Jeongsoo Choi, Ji-Hoon Kim, Shujie Hu, and Joon Son Chung. Prosocodec: Prosody-oriented speech codec for voice conversion. *arXiv preprint arXiv:2606.21888*, 2026.

Thierry Dutoit, Vincent Pagel, Nicolas Pierret, François Bataille, and Olivier Van der Vrecken. The mbrola project: Towards a set of high quality speech synthesizers free of use for non commercial purposes. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, volume 3, pp. 1393–1396. IEEE, 1996.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.

Haorui He, Zengqiang Shang, Chaoren Wang, Xuyuan Li, Yicheng Gu, Hua Hua, Liwei Liu, Chen Yang, Jiaqi Li, Peiyang Shi, et al. Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pp. 885–890. IEEE, 2024.

Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, et al. Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models. *arXiv preprint arXiv:2403.03100*, 2024.

Wei Kang, Xiaoyu Yang, Zengwei Yao, Fangjun Kuang, Yifan Yang, Liyong Guo, Long Lin, and Daniel Povey. Libriheavy: A 50,000 hours asr corpus with punctuation casing and context. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 10991–10995. IEEE, 2024.

Sotirios Karapiperis, Nikolaos Ellinas, Alexandra Vioni, Junkwang Oh, Gunu Jho, Inchul Hwang, and Spyros Raptis. Investigating disentanglement in a phoneme-level speech codec for prosody modeling. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pp. 668–674. IEEE, 2024.

Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchiani, Yu Zhang, Wei Han, and Ankur Bapna. Libritts-r: A restored multi-speaker text-to-speech corpus. *arXiv preprint arXiv:2305.18802*, 2023.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan. *Advances in neural information processing systems*, 36:27980–27993, 2023.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. Bigvgan: A universal neural vocoder with large-scale training. *arXiv preprint arXiv:2206.04658*, 2022.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Matthias Mauch and Simon Dixon. pyin: A fundamental frequency estimator using probabilistic threshold distributions. In *2014 ieee international conference on acoustics, speech and signal processing (icassp)*, pp. 659–663. IEEE, 2014.

Lingwei Meng, Long Zhou, Shujie Liu, Sanyuan Chen, Bing Han, Shujie Hu, Yanqing Liu, Jinyu Li, Sheng Zhao, Xixin Wu, et al. Autoregressive speech synthesis without vector quantization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1287–1300, 2025.

Max Morrison, Cameron Churchwell, Nathan Pruyne, and Bryan Pardo. Fine-grained and interpretable neural speech editing. *arXiv preprint arXiv:2407.05471*, 2024.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

Kaizhi Qian, Yang Zhang, Shiyu Chang, Mark Hasegawa-Johnson, and David Cox. Unsupervised speech decomposition via triple information bottleneck. In *International Conference on Machine Learning*, pp. 7836–7846. PMLR, 2020.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pp. 28492–28518. PMLR, 2023.

Franck Ramus and Jacques Mehler. Language identification with suprasegmental cues: A study based on speech resynthesis. *The Journal of the Acoustical Society of America*, 105(1):512–521, 1999.

Gerit P Sonntag and Thomas Portele. Purra method for prosody evaluation and investigation. *Computer Speech & Language*, 12(4):437–451, 1998.

Guangzhi Sun, Yu Zhang, Ron J Weiss, Yuan Cao, Heiga Zen, and Yonghui Wu. Fully-hierarchical fine-grained prosody modeling for interpretable speech synthesis. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6264–6268. IEEE, 2020.

Antti Suni, Juraj Šimko, Daniel Aalto, and Martti Vainio. Hierarchical representation and estimation of prosody using continuous wavelet transform. *Computer Speech & Language*, 45:123–136, 2017.

Hui Wang, Siqi Zheng, Yafeng Chen, Luyao Cheng, and Qian Chen. Cam++: A fast and efficient network for speaker verification using context-aware masking. *arXiv preprint arXiv:2303.00332*, 2023.

Lukas Wolf, Tiago Pimentel, Evelina Fedorenko, Ryan Cotterell, Alex Warstadt, Ethan Gotlieb Wilcox, and Tamar I Regev. Quantifying the redundancy between prosody and text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9765–9784, 2023.

A Implementation Details

A.1 TTS target generation

The initial template is synthesized with MBROLA us3: interior vowels map to |i|, consonants to |s|, and the first and last interval labels remain unchanged.

We train the first F5 model for 400k updates with $M(d)$ as the acoustic condition, then initialize the second model from its checkpoint and fine-tune it for 50k updates with $M(d_1)$ as the acoustic condition. Both models use true English phones and CampPlus embeddings from the original utterance to predict $M(x)$. We obtain the phones using Phonemizer (Bernard & Titeux, 2021) with eSpeak NG[2].

For target generation, the first model uses the template phone sequence to produce d_1; the second uses all-|m| phones to produce the codec-training target τ. Both use 32 flow steps.

A.2 Codec training

The codec uses a DAC encoder and decoder with 1,024-dimensional latent channels. Encoder strides are $[2, 4, 8, 8]$, and the decoder mirrors them with rates $[8, 8, 4, 2]$. Each of the nine codebooks has code dimension 8 and 1,024 entries.

All nine codebooks are used during evaluation and the 100k continuation. Both codec stages use AdamW with a base learning rate of 10^{-4}, $\beta = (0.8, 0.99)$, weight decay 0.01, and exponential decay 0.999996 per update. Codec-generator loss weights are 15 for multi-scale mel reconstruction, 2 for discriminator feature matching, 1 for adversarial loss, 0.25 for commitment loss, and 1 for codebook loss. We train on fixed 3-second crops.

A.3 Downstream speech generator

The speech-generator transformer has 22 layers, hidden width 1,024, 16 attention heads, and feed-forward width 4,096. It is trained with conditional flow matching for 400k updates using AdamW, learning rate 7.5×10^{-5}, $\beta = (0.9, 0.98)$, weight decay 0.01, and 20k warmup updates followed by inverse-square-root decay. Mel features use 16 kHz waveforms, 1,024-point FFT and window, 320-sample hop, and 80 bins. The external speaker condition is a 192-dimensional embedding.

All three systems use their final 400k checkpoints and a shared sampling seed. The shared speaker embedding comes from a fixed, different utterance by the same speaker as the source. The vocoder uses the BigVGAN-v2 implementation.[3]

A.4 Matched F0+Energy system

The matched F0+Energy condition is computed online from the peak-normalized 16 kHz source waveform with centered 1,024-sample Hann frames and a 320-sample hop. The first channel is derived by estimating F0 within 50–600 Hz, taking its logarithm, and mapping it to $[-1, 1]$; frames failing the configured voicing or energy test are assigned -2 to mark unvoiced frames. For the second channel, we compute Hann-windowed RMS energy, clip it to $[10^{-5}, 1]$, take its logarithm,

[2] https://github.com/espeak-ng/espeak-ng

[3] https://github.com/NVIDIA/BigVGAN

and map it to $[-1, 1]$ using the same fixed normalization for every utterance. A learned linear projection maps the two channels to the 80-dimensional F5 acoustic-condition space.

A.5 FACODEC FRONT END

The FACodec condition uses the official frozen encoder and first residual quantizer[4], corresponding to the prosody stream. Its 256-dimensional dequantized vectors use a native hop of 200 samples and are sampled at the nearest frame centers on the F5 hop-320 grid, then linearly projected to 80 dimensions. Content, timbre, and residual codes are not supplied to F5.

A.6 RULE-BASED CONTROLS

For the direct all-ɨmɨ rule control, every non-silent phone is replaced by ɨmɨ while source interval durations and F0 are retained.

We generate MBROLA targets by randomly replacing phones within the same class, using only phone pairs supported by MBROLA. We train the same codec from scratch for 250k updates to map original speech to these targets, using aligned 3 s crops. All training targets are MBROLA-generated.

A.7 CTC LEXICAL PROBES

Each recognizer has 12 layers, hidden size 256, four attention heads, feed-forward width 2,048, convolution kernel 15, and $2\times$ subsampling. Training uses AdamW with learning rate 4×10^{-4}, 25k warmup steps, and dropout 0.1. We evaluate the original-speech model at 165k updates and the other two at 200k, using greedy decoding without a language model.

B METRIC IMPLEMENTATION DETAILS

B.1 FRAME-LEVEL PROSODY.

We compare pitch, energy, and voicing in each waveform with the corresponding original speech at the same time positions. F0 is estimated with pYIN (Mauch & Dixon, 2014)[5]. Reference F0 frames are voiced in both the original speech x and its vocoder reconstruction x_{voc}. F0 recovery is the proportion of these reference frames with detectable F0 in the evaluated waveform. On the recovered frames, F0 MAE measures the mean absolute pitch difference in cents, while Pearson correlation between log-F0 tracks measures how closely their pitch changes follow each other.

With a nonempty reference but no recovered frames, F0 recovery is zero and pitch accuracy is undefined. MAE requires at least one recovered frame; correlation requires at least three and nonzero variance in both tracks.

Energy correlation compares mean-centered log-RMS contours. VUV agreement measures how often the two waveforms have the same voiced/unvoiced state. Both use all common frames. F0 recovery pools frame counts across utterances; the other scores are calculated per utterance and summarized by their median.

B.2 PHONE TIMING.

We compare the durations and timing of corresponding phones in generated and original speech. A CTC model trained on original speech aligns the same reference text to both waveforms on a 40 ms grid. Symbols belonging to one eSpeak phone, including stress marks, are grouped into a single phone.

We compare corresponding phone durations and boundaries between adjacent phones within a word. Each boundary is the midpoint between the preceding phone's end and the following phone's start. Absolute errors are averaged within each utterance, then summarized by their median in milliseconds.

[4] https://huggingface.co/amphion/naturalspeech3_facodec

[5] https://librosa.org/doc/0.11.0/generated/librosa.pyin.html

B.3 MULTI-SCALE PROSODY.

WPT[6] measures whether a waveform preserves both fast and slow prosodic changes in the original speech. It combines standardized interpolated F0, energy, and an energy-derived speaking-rate curve with weights of 1, 1, and 0.5. The combined curve is detrended and normalized. Wavelet analysis compares its variation at periods of 0.125, 0.25, 0.5, and 1 s. Comparisons use the original time positions and exclude one period at each endpoint. We report median utterance-level correlations at each scale.

B.4 AUTOPCP.

AutoPCP[7] measures prosodic similarity between generated and original speech. We use the official multilingual-v2 model. Each pair is scored in both input orders; the two scores are averaged before taking the mean across utterances.

B.5 CTC LEXICAL RECOVERY.

We compare the CTC-decoded sequence from each representation with the reference phone sequence to test whether the original words can be recovered. PER divides the total number of substitutions, deletions, and insertions by the total reference length, counting eSpeak symbols including stress marks. An utterance is error-free when there are no edits. A word is recovered only when its complete reference symbol sequence matches the aligned recognition output.

B.6 WHISPER DECODING.

We use Whisper large-v3[8] to transcribe reconstructed speech and compare the transcription with the original text. English WER pools word errors and reference words across utterances after `EnglishTextNormalizer` normalization. For codecs trained on different targets, we also count outputs whose transcriptions contain at least three or five consecutive words from the original text.

Cross-lingual decoding supplies the language but not the reference text. Mandarin uses character error rate after removing spaces, punctuation, and other symbols; English and German use WER after case folding and punctuation removal.

B.7 SPEAKER MEASUREMENTS.

We compare whole-utterance CampPlus embeddings[9] of each waveform and its original speech using cosine similarity. SIM is the median similarity across utterances.

For speaker identification, each of the 40 speakers is represented by the mean of their normalized original-speech embeddings. This mean excludes the original recording corresponding to the waveform being evaluated. We select the speaker whose mean embedding has the highest cosine similarity to the evaluated waveform's embedding.

B.8 PHONE SWITCHING.

We test whether changing the phone input shifts the generated pronunciation toward the requested phone while preserving source prosody. The CTC recognizer trained on original speech decodes the generated waveforms. We report the ˌmˌ and ˌsˌ shares separately, each as its count divided by the total number of decoded nonblank symbols across utterances. PER compares the decoded sequence with the original phone sequence.

[6] https://github.com/asuni/wavelet_prosody_toolkit

[7] https://github.com/facebookresearch/stopes/tree/main/stopes/eval/auto_pcp

[8] https://huggingface.co/openai/whisper-large-v3

[9] https://huggingface.co/FunAudioLLM/Fun-CosyVoice3-0.5B-2512/blob/main/campplus.onnx

Prosody is compared with original speech using the frame-level metrics above; F0 is not scored for all-ɪsl. Supporting measures are the fraction of frames with periodic voicing detected by autocorrelation and the fraction of spectral energy above 4 kHz. Both are summarized by utterance-level medians.

B.9 Local pitch effects.

We first synthesize speech from unmodified DelexWav, then raise or lower the pitch in the middle 35–65% interval of DelexWav and synthesize again, keeping the text and speaker conditions unchanged. We compare pitch at the same time positions in the two generated versions, separately for the selected interval and the rest of the utterance.

Pitch changes are measured in cents where both versions have detectable F0. We take the median change in each region, then the median across utterances. Directional success is the fraction of measurable utterances whose pitch change within the selected interval has the requested sign.

B.10 Local duration control and phone timing.

We synthesize speech before and after lengthening or shortening the middle section of DelexWav, keeping the text and speaker conditions unchanged. We test how much the corresponding speech lengthens or shortens and whether speech outside that interval retains its duration before editing.

CTC alignments locate corresponding speech segments in the two versions. We total their durations separately within the selected interval and in the rest of the utterance. The ratio of edited to unedited totals measures the duration change: for example, 1.10 means a 10% increase. Segment midpoints in the unedited version determine which region they belong to. Ratios are calculated per utterance and summarized by their median. The change has the requested direction when the ratio within the selected interval is above one for expansion or below one for compression.

We also check how accurately each phone follows the requested time scaling. Expected positions remain unchanged before the selected interval, scale proportionally within it, and shift together after it. We use this mapping to calculate each phone's expected duration and boundary, then compare them with the speech generated after editing. Errors follow the phone-timing calculation above, with regions determined from unedited phone midpoints and boundaries.

B.11 Prosody and content after time stretching.

We test whether changing duration preserves the pitch and energy patterns of the speech generated before editing, and whether it increases word errors.

Because the two versions differ in length, spectral DTW first identifies corresponding time positions. The aligned positions must remain within 0.50 s of the positions expected under the requested time scaling. Pitch and energy use the same alignment. Within the selected interval and the rest of the utterance, we measure F0 recovery, F0 MAE, and energy correlation. Here, F0 recovery uses the voiced frames in the unedited speech as its reference and pools frame counts across utterances. F0 MAE uses recovered reference frames, while energy correlation uses all aligned frames; both are summarized by utterance-level medians.

We separately compare the edited speech's transcription with the original text and report the change in pooled WER from before editing, in percentage points. The same evaluation applies to DelexWav editing, F0+Energy editing, and conventional speech time stretching, each compared with its own unedited speech.

C Additional Results

C.1 Multi-scale prosody in direct waveforms

Table 6 evaluates pitch and energy separately, whereas Table 1 combines the pitch, energy, and energy-derived speaking-rate curves before computing correlations.

Table 6: WPT component correlations for direct waveforms. Values are medians. Higher is better.

Waveform	Component	0.125 s	0.25 s	0.50 s	1.00 s
MBROLA /s/→/i/	F0	0.35	0.52	0.69	0.82
MBROLA /s/→/i/	Energy	0.55	0.64	0.71	0.83
MBROLA /m/	F0	0.52	0.68	0.82	0.90
MBROLA /m/	Energy	0.20	0.31	0.54	0.77
DELEXWAV	F0	0.48	0.71	0.85	0.92
DELEXWAV	Energy	0.64	0.72	0.76	0.84

C.2 CROSS-LINGUAL DECODING DIAGNOSTICS

Emilia utterances span multiple speakers and durations of 3–10 s. BigVGAN generates only the reference x_{voc}; DELEXWAV is produced by the codec's own decoder. As shown in Table 7, ordinary Whisper decoding has much higher source-text error on the extracted waveforms than on original speech. These off-the-shelf decodes are separate from the representation-trained English CTC probes.

Table 7: Auxiliary Whisper decoding on 100 Emilia utterances per language. Mandarin uses CER; English and German use WER. Rates are pooled, including empty hypotheses.

Language / metric	Reference units	Original speech	DELEXWAV
English / WER	1,951 words	7.53%	120.66%
Mandarin / CER	3,117 characters	16.33%	111.77%
German / WER	1,725 words	12.12%	108.99%

C.3 PROSODY IN RECONSTRUCTED SPEECH

As shown in Table 8, F0+Energy yields higher WPT correlations than DELEXWAV, with similar AutoPCP scores.

Table 8: Generated-speech WPT combined correlations and mean AutoPCP, all over 1,234 utterances. Higher is better.

Output	0.125 s↑	0.25 s↑	0.50 s↑	1.00 s↑	AutoPCP↑
x_{voc}	0.98	0.99	0.99	0.99	4.02
F0+Energy + TTS	0.95	0.96	0.97	0.98	3.24
FACodec + TTS	0.77	0.84	0.89	0.93	3.03
DELEXWAV + TTS	0.87	0.91	0.93	0.96	3.25

C.4 RECOGNIZER DECODES OF FIXED-PHONE OUTPUTS

As shown in Table 9, all-/m/ and all-/s/ inputs increase the presence of the requested phone in the decoded sequences, although the recognizer still predicts other phones.

C.5 EXPLORATORY CROSS-TEXT DIAGNOSTIC

The cross-text diagnostic uses 48 A/B pairs and three matched seeds, giving 144 outputs with A's DELEXWAV waveform and speaker embedding but B's text. The waveform, speaker embedding, and seed are fixed when comparing the two texts. An output is closer to B when its normalized WER to B is lower than its WER to A; the reverse defines closer to A, and equal values are ties. Corpus WER is 101.89% to A and 103.97% to B. In total, 59 outputs are closer to B, 75 are closer to A, and 10 are ties, with none exactly matching B. This diagnostic does not support reliable cross-text synthesis.

Table 9: Recognizer decodes for three utterances under source-phone, ˌmˌ, and ˌsˌ inputs.

Sample	Phone input	Decoded sequence
134686-0026	GT	ðə ɹˈɛktɚ dɪdnˌɑːt ˈæsk fɚɹə kˈæɾɪdʒˌɪzəm tə hˈɪɹ ðə lˈɛsən fɹʌm
	ˌmˌ	ðɪ mˈɛv bˈi maɪ mˈæm fˈɔː ʌv mˈɪm ɚæn bɨnˌiːn ðə mˈænz lˈʌ
	ˌsˌ	tə sˈɪst ɪt əsdˈæst t ə tʃˈɪst tˈɪsɪstəsˈɪːs pə sˈɛsɛstɚz
134686-0037	GT	ɪnðə sˈaɪləns ðɛɹ dˈɑːɹk fˈaɪɚ kˈɪndəl ʌvðə dˈʌsk ˌɪntʊ ɐ tˈɔːni ɡlˈoʊ
	ˌmˌ	mˈæn mˈɛt ðɛɹ pfˈeɪ lˈɪmdəl ðˌæn ə mˈʌ ɪn fɹʌm ʌv mˈɑːɹnɪn mˈɑː
	ˌsˌ	ðə sˈædʒɪt tˈɪns ðɛɹ sˈeɪz tsˈuːsɨz ʌv ðə tˈɛsk ɪz tə sʌtsˈɑːldʒɪts ɐbˈaʊt
134686-0015	GT	bˌʌt ðə dˈʌsk dˈiːpənɪŋ ɪnðə skˈuːlɹuːm kˈʌvɚd ˌoʊvɚ hɪz θˈɔːts ðə bˈɛl ɹˈæŋ
	ˌmˌ	ðˈɛn ðə mˈʌn nˈɛv æn ðə mˈɔnəm mˈʌn ɹˈoʊmʌ mɪm ˈaʊpɪm ðə mˈɛʊ mˈæn
	ˌsˌ	ɪˈʌst dðɪstəz ʌ ðə stɹˈoʊsɪstɪz plˈɑːzɪɾ ʌðɪ stˈæts ðətˈɜːst lˈeɪɾɨ

C.6 Time-stretch preservation and timing

As shown in Table 10, F0+Energy and conventional waveform stretching preserve inside-region contours more closely than editing DelexWav.

Table 10: Edited-region contour preservation on 200 utterances per condition. F0 MAE is in cents, energy is correlation, recovery and WER are percentages, and ΔWER is in percentage points.

Method	Scale	F0 MAE↓	Energy↑	F0 rec.↑	WER↓	ΔWER↓
DelexWav	1.25	68.05	0.86	76.71	20.61	+10.26
F0+Energy	1.25	37.61	0.96	91.09	45.96	+8.98
Waveform stretch	1.25	40.00	0.95	88.86	11.54	+1.20
DelexWav	0.80	68.95	0.82	75.46	24.82	+14.47
F0+Energy	0.80	39.92	0.96	86.65	45.82	+8.84
Waveform stretch	0.80	51.89	0.93	84.09	15.00	+4.66

Outside-region energy correlations are 0.98/0.98 for DelexWav, 1.00/1.00 for F0+Energy, and 1.00/1.00 for waveform stretching; corresponding F0 errors are 29.66/29.86, 21.73/23.14, and 8.87/9.50 cents.

Table 11 reports phone-level timing errors against the prescribed time transformation. F0+Energy has slightly smaller inside-region errors. DelexWav has smaller outside-region boundary errors, while outside duration errors are mixed.

Table 11: Phone-level timing errors relative to the prescribed local time map, in ms. Every cell uses all 200 utterances; lower is better.

Method	Scale	Inside dur.↓	Inside bound.↓	Outside dur.↓	Outside bound.↓
DelexWav	1.25	21.91	54.76	7.09	18.98
F0+Energy	1.25	21.11	54.08	7.02	28.60
DelexWav	0.80	15.25	49.29	7.24	23.48
F0+Energy	0.80	14.69	47.55	7.56	31.60

D Construction analysis

We examine how target construction and further codec training affect the extracted waveform.

The fixed-phone TTS target route yields fewer recoverable source-word fragments than the random-phone rule route. As shown in Table 12(a), Whisper recovers at least three consecutive source words in 433 outputs of the MBROLA-target codec and none of the TTS-target codec; the corresponding five-word counts are 125 and 0. Both codecs are evaluated at 250k updates.

Speaker-conditioned TTS targets retain more of the original speaker's voice than fixed-voice MBROLA. As shown in Table 12(b), TTS targets have higher similarity to the original speech (0.65 vs. 0.15), while the final DELEXWAV waveform scores 0.45. This voice retention motivates the TTS targets, although their pitch and energy contours are less accurate than those of the all-ɪmɪ rule waveform (Table 12(c)).

Further training with original-speech targets. As shown in Table 12(c), codec training improves pitch and energy preservation relative to the TTS targets. Training for another 100k updates with 10% original-speech targets reduces median F0 MAE from 57.85 to 51.96 cents and slightly improves energy correlation and voicing agreement. F0 recovery decreases from 92.38% to 90.96%.

As shown in Table 12(d), the CTC recognizer trained on the 30% version recovers 14,941 of 21,284 reference words, while the recognizer trained on the 10% version recovers none. These results support the 10% setting: it improves several acoustic measures while maintaining strong lexical suppression.

Table 12: Comparison of training targets and codec outputs on 1,234 test utterances. (a) compares two codecs trained for 250k updates; (b) compares fixed-phone waveforms; (c) compares rule waveforms, TTS targets, and codec outputs before and after further training with 10% original-speech targets; (d) compares lexical recovery from the 10% and 30% versions.

(a) Source-word fragments recovered by Whisper.

Codec target	≥ 3 source words↓	≥ 5 source words↓
Random-phone MBROLA	433 / 1,234	125 / 1,234
TTS fixed-phone	0 / 1,234	0 / 1,234

(b) Speaker similarity and identification.

Waveform	Cosine similarity↑	Speaker identification↑
MBROLA ɪmɪ	0.15	3.89%
TTS fixed-phone target	0.65	84.93%
DELEXWAV	0.45	38.17%

(c) Acoustic comparison of targets and codec outputs.

Waveform	F0 rec.↑	F0 MAE↓	F0 corr.↑	Energy corr.↑	VUV↑
MBROLA ɪmɪ	92.88%	37.20	0.97	0.67	0.69
TTS fixed-phone target	91.63%	93.80	0.87	0.62	0.66
Codec, 250k (TTS targets only)	92.38%	57.85	0.95	0.84	0.70
DELEXWAV, 350k (10% original targets)	90.96%	51.96	0.96	0.85	0.71

(d) Lexical recovery with 10% and 30% original-speech targets.

Original-speech targets	PER↓	Recovered words↑
10% (DELEXWAV)	92.65%	0 / 21,284
30%	13.08%	14,941 / 21,284