COOKVOICE: AN UNIFIED FRAMEWORK FOR STYLE CONTROLLABLE MULTI-MODAL HUMAN VOICE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Human voice generation has made rapid progress in speech generation, singing voice generation, voice cloning, and voice editing. However, most existing systems are designed for specific tasks and often rely on task-dependent architectures, control signals, or autoregressive decoding, limiting fine-grained controllability and inference efficiency. In this paper, we propose **CookVoice**, a unified framework for multimodal, multi-style, and multi-task human voice generation. CookVoice decomposes the human voice into three key factors: content, prosody, and style, enabling both speech and singing voice generation within a unified model. To achieve precise and flexible controllability, we design a flexible alignment strategy that maps text, style, and prosody control signals onto the frame-level of spectrogram. This design allows CookVoice to support a wide range of tasks, including text-to-speech, text-to-singing voice, style-controllable generation, voice mimicry, voice conversion, and voice editing. Experimental results show that CookVoice achieves generation quality competitive to existing Text-to-Speech and text-to-singing voice baselines, while providing stronger style and prosody controllability. Moreover, CookVoice achieves competitive performance to large-scale baselines with only 43.51 million parameters and efficient inference using as few as **4** ODE steps, making it a practical solution for real-world human voice generation applications. Anonymous demo page available here.

1 INTRODUCTION

Recent years have witnessed an emergence of human voice generative AI, with models capable of high-quality text-to-speech (TTS) Du et al. (2024); Deng et al. (2025); Lou et al. (2025a); Chen et al. (2025), singing voice generation Liu et al. (2022); Zhang et al. (2024a;b), zero-shot voice cloning Deng et al. (2025); Chen et al. (2025); Du et al. (2024), and voice editing Zhang et al. (2026). Driven by advances in acoustic representation learning Kong et al. (2020); Lou et al. (2026b), and generative artificial intelligence Peebles & Xie (2023); Lipman et al. (2022) modern systems can generate increasingly natural and expressive voices from multi-modal input. However, despite this progress, most existing voice generation systems are still designed around isolated tasks. TTS models focus on generating speech from text Ren et al. (2019), singing voice synthesis models typically require musical scores and note input Zhang et al. (2022); Liu et al. (2022), style-controllable TTS models rely on text style prompts or reference speech Lou et al. (2025a); Du et al. (2024), and voice cloning models mainly emphasize speaker similarity Chen et al. (2025). Different forms of human voice generation tasks are commonly treated as separate problems, requiring task-specific architectures, datasets, training objectives, and control signals.

This fragmentation makes human voice generation unnecessarily complex. Theoretically, speech and singing share the same fundamental linguistic basis, yet they differ substantially in temporal structure and prosodic behavior. In speech, duration and prosody are often implicit and context-dependent, while in singing, they are explicitly constrained by musical notes, beats, and melody. Moreover, style-controllable voice generation focuses on cloning the rich paralinguistic styles within reference speech, such as emotion, gender, age, timbre, and speaking or singing style. These styles may come from different modalities, including natural language descriptions, reference speech, reference singing voices, lexical tones, stresses, notes, or F0 contours. Existing methods usually handle

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.

only a subset of these controls, making it difficult to flexibly decide *what* to generate, *how* it should sound, and *where* its prosody should come from.

A unified model capable of generating different forms of human voice under a single controllable interface is highly desirable for both academia and the industrial community. Ideally, such a model should support both speech and singing voice generation, accept multimodal style control, and allow explicit control over prosody. This requires a framework that can represent human voice generation through shared factors, including content, prosody, and style, while preserving the distinct characteristics of different tasks. More importantly, the framework should remain simple to train, straightforward to use, and efficient at inference time.

However, achieving such a unified framework remains challenging. Many recent voice generation models rely on autoregressive (AR) decoding or implicit alignment mechanisms Du et al. (2024); Deng et al. (2025), while recent unified voice generation systems further attempt to cover multiple tasks within a single model Zhang et al. (2025; 2026). Although these models can achieve strong generation quality, their controllability is often constrained by the underlying generation mechanism. AR models generate acoustic tokens sequentially and therefore model duration and prosody implicitly through token prediction. This design is flexible at the utterance level, but it makes fine-grained temporal control difficult, since users cannot directly specify how each phoneme, note, or frame should evolve over time. In contrast, many NAR systems Ren et al. (2019); Lou et al. (2025a); Kim et al. (2021); Chen et al. (2025) improve temporal stability by imposing explicit or restricted alignments, such as phoneme-level durations Ren et al. (2019); Kim et al. (2021); Lou et al. (2025a), note-to-phoneme mapping Zhang et al. (2022; 2024a;b), or sentence-level prosody references Zhang et al. (2025; 2026). While these designs improve controllability over AR decoding, they often rely on predefined assumptions between content and prosody, which limits their flexibility when multiple control signals are jointly involved.

These limitations become more severe for tasks requiring precise temporal control, such as singing voice generation and prosody mimicry. For singing, phonemes must follow score-defined note durations and melodies; for voice mimicry, the generated voice should preserve detailed frame-level prosodic patterns from a reference signal. When content, style, and prosody are combined, implicit AR alignment or restricted NAR alignment makes it difficult to achieve both flexible control and precise temporal consistency. To address this issue, CookVoice adopts a flexible frame-level alignment strategy. Instead of assuming a fixed dependency between content and prosody, CookVoice explicitly expands different control signals to the acoustic frame level and allows them to be combined freely. This design enables more fine-grained and flexible controllability across speech, singing voice generation, and prosody mimicry.

In this work, we propose **CookVoice**, a unified non-autoregressive (NAR) framework for multimodal, multi-style, and multi-task human voice generation. CookVoice formulates human voice generation as conditional latent acoustic generation, where the target voice is represented in a compact latent space and generated from three shared factors: content, prosody, and style. Content is represented by phoneme sequences derived from text or lyrics. Prosody can be controlled through discrete signals, such as lexical tones, stresses, and MIDI notes, or continuous frame-level F0 contours extracted from a reference voice. Style can be provided by either text-based descriptions or reference voice prompts, enabling both natural language style control and reference-based style transfer.

Instead of relying on implicit AR token-prediction, CookVoice adopts temporally aligned NAR generation pipeline which explicitly align content, prosody, and style tokens to acoustic tokens at frame level. For speech, phoneme durations are learned or predicted from alignment information. For singing voice, durations are deterministically derived from musical score timing. This design allows speech and singing voices to be modeled under the same temporal framework, while preserving their different duration and prosody structures. The temporally aligned representations are then integrated through a multimodal adaptive fusion module and passed to a flow-matching Diffusion Transformer for high-quality latent acoustic generation.

CookVoice supports a broad range of human voice generation tasks within a single framework, including TTS, text-style-controllable TTS, voice-style-controllable TTS, singing voice generation, style-controllable singing voice generation, prosody mimicry and many others. These tasks are handled by activating different combinations of content, prosody, and style conditions, rather than

training separate task-specific models. As a result, CookVoice is a simple, interpretable, and highly controllable framework for the generation of human voice in both speech and singing domains. The main contributions are summarized as follows:

1. We propose **CookVoice**, a unified framework for multi-task human voice generation. By decomposing human voice into style, content, and prosody, CookVoice successfully unifying multiple voice generation task within a unified model.

2. We design a flexible alignment mechanism for fine-grained controllable human voice generation. By aligning different control signals at frame level, CookVoice can flexibly support multimodal style and prosody control, enabling more precise control over paralinguistic style, melody, duration, and prosodic expression.

3. Our experiment demonstrate CookVoice achieves higher style and prosody controllability than existing baselines, including models trained with larger-scale data or larger model sizes. Despite using a lightweight architecture and a much smaller training data, CookVoice achieves superior style similarity and F0 controllability while maintaining efficient inference and competitive generation quality.

2 PRELIMINARY

Human Voice Generation (HVG) is to generate waveform from different modality of control signal. In this study, we define the human voice as a combination of human speech and the singing voice. Control signals include **text**, **lyric**, **note**, and **reference voice**. Full details covered tasks are presented in Appendix H. For readers unfamiliar with linguistic and musical concepts, we provide terminology definitions in Appendix I.

Formally, we formulate HVG tasks as follow. Let $Y \in \mathbb{R}^{N \times T}$ denotes the latent embedding extracted from a spectrogram via HiFi-GAN Kong et al. (2020), with N and T denoting the latent dimension and the number of frames. To systematically model the generation process, we assume the human voice can be disentangled into five independent key components:

Style ($S \in \mathbb{R}^D$): Encodes a global paralinguistic styles embedding of voice, indicating how the voice is expressed (e.g., gender, age, and emotion). This style embedding can be conditioned on different sources: either a text description $S_{\mathcal{T}}$ (e.g., *"A female speaking happily"*), or a reference voice $S_{\mathcal{V}}$ to clone the speaking style of target voice $\mathcal{V}$;

Content ($X \in \mathbb{R}^{L_1}$): Represents the linguistic content to be voiced. Content in Text-to-Speech (TTS) can be text, while in Text-To-Singing Voice (TTSV), it refers to the lyrics. In this study, X specifically denotes the sequence of phonemes;

Prosody ($\mathcal{P}$): contains all forms of prosody variations that shape the expressiveness and melody of the voice. We categorize prosody signals into discrete and continuous signals. The discrete signal ($\mathcal{P}_{disc}$) contains lexical prosody tokens $\mathcal{P}_{lex} \in \mathbb{R}^{L_1}$ (e.g., tones in Mandarin Chinese or word stress in English) for speech, and musical notes $\mathcal{P}_{note} \in \mathbb{R}^{L_2}$ for singing voices. While musical notes govern prosody variation similar to lexical tones, their sequence length (L_2) can differ from the phoneme length (L_1), as a single musical note may span multiple phonemes, or a single phoneme may stretch across multiple notes. Alternatively, the continuous prosody ($\mathcal{P}_{cont}$) represents the frame-level fundamental frequency ($F_0 \in \mathbb{R}^T$) contour extracted from a voice. It aligns temporally with the latent embedding, sharing the same frame length T.

Our objective is to train a generative model $\mathcal{G}(\cdot)$ that can generate latent embedding $\hat{Y}$ conditioned on different source of control signal ($\hat{Y} = \mathcal{G}(S, X, \mathcal{P})$). Depending on the requirements of different tasks, S and $\mathcal{P}$ can be selectively conditioned on different source. Detailed task formulations are provided in Table 3.

For the latent embedding, we encode the spectrogram into a compact continuous latent space using a HiFi-GAN-style Kong et al. (2020) autoencoder (AE), with further details provided in Appendix B. Once the AE is fully trained, its parameters are frozen, and the latent embedding Y is used as the optimization target for CookVoice during training.

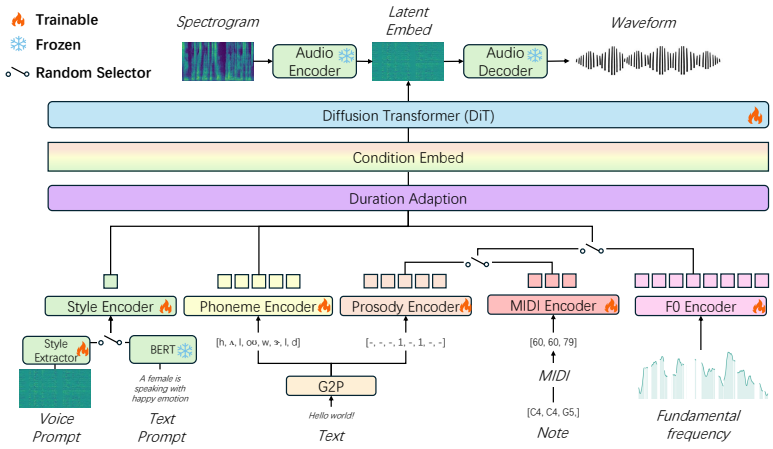


Figure 1: Architecture of the CookVoice

3 CookVoice

The overall architecture of CookVoice is presented in Figure 1. This section will present the architecture of CookVoice in detail.

3.1 Style Encoder

Style Encoder is designed to generate a style embedding S, which dictates the stylistic characteristics of the generated voice. In CookVoice, style conditioning can be derived from two modalities.

The first modality is text, coming from the text-based style prompt $S_{\mathcal{T}}$. In this scenario, the style is controlled by natural language descriptions (e.g., *a female is speaking with a happy emotion*). Given a text-based style description $\mathcal{T}$, we first extract its semantic embedding using a pre-trained, frozen, MPNet Song et al. (2020) sentence encoder. The resulting embedding is then projected through a trainable linear layer to obtain the text style embedding $S_{\mathcal{T}} \in \mathbb{R}^D$.

The second modality is a reference voice, $\mathcal{V}$. This addresses scenarios requiring zero-shot voice cloning or timbre transfer, where the generated voice must mimic a reference voice. We encode the acoustic signal into a latent space using the encoding method described in Section B. The resulting latent embedding $\mathcal{V} \in \mathbb{R}^{N \times T}$ is then processed by a trainable Transformer Vaswani et al. (2017) encoder with an attentive pooling layer to obtain the voice style embedding $S_{\mathcal{V}} \in \mathbb{R}^D$.

During model training, the overall style condition S is uniformly sampled from either the text modality $S_{\mathcal{T}}$ or the voice modality $S_{\mathcal{V}}$ with a 50% probability for each item within the batch. To align with the temporal resolution of the latent embedding $\mathcal{V} \in \mathbb{R}^{N \times T}$, the style embedding S is expanded by repeating it T times along the temporal axis, producing final style embedding $S_e \in \mathbb{R}^{D \times T}$.

3.2 Content Encoder

We employ a LanStyleTTS multilingual Grapheme-to-phenome (G2P) module Lou et al. (2025b) to convert text into phonemes X and prosody tokens P, where $X, P \in \mathbb{R}^{L_1}$. The phoneme sequence X captures the linguistic content, while the prosody sequence P explicitly models phoneme-level prosodic variations (e.g., lexical tones in Mandarin or phoneme stress in English) across different linguistic contexts. In CookVoice, X strictly denotes the linguistic content, while P serves as a discrete prosody-conditioning signal (also detailed in Section 3.3).

Phonemes are first mapped into a sequence of embeddings $X \in \mathbb{R}^{D \times L_1}$ and then processed through Feed-Forward Transformer (FFT) blocks Ren et al. (2019). A critical factor for natural and expressive voice generation is the duration of the phoneme $\mathcal{D}_X \in \mathbb{Z}^{L_1}$, which exhibits vastly different behaviors in the speech versus singing voice. In conventional TTS tasks, the absence of explicit duration constraints necessitates the use of a duration predictor Ren et al. (2019); Lou et al. (2025a)

or AR architectures Du et al. (2024); Wang et al. (2017) for implicit inference. Conversely, in TTSV tasks, note durations are explicitly dictated by musical scores. To unify both tasks within CookVoice, we adopt a flexible duration alignment and expansion strategy.

During the training stage, we handle speech and singing voice differently. For speech, we use pre-trained ParaStyleTTS Lou et al. (2025a) to find optimal alignments between the speech audio and the phoneme sequence, yielding ground-truth durations $\mathcal{D}_X$. For singing voice, where score durations are denoted in relative beats rather than absolute time (e.g., seconds), we compute the relative temporal proportion of each phoneme based on its assigned beats. This relative proportion is then multiplied by the total number of acoustic frames T to ascertain the absolute frame-level duration. Both duration sequences satisfy the constraints $\mathcal{D}_X \in \mathbb{Z}^{L_1}$ and $\sum_{i=1}^{L_1} \mathcal{D}_X^i = T$. Each phoneme embedding X_i is then repeated $\mathcal{D}_X^i$ times to produce the duration-expanded content embedding $X_e \in \mathbb{R}^{D\times T}$, which perfectly matches the temporal dimension of the latent acoustic embedding.

During the inference stage, we use a trained duration predictor from ParaStyleTTS Lou et al. (2025a) to predict phoneme durations for bilingual speech. For singing voice generation, the duration is deterministically computed from the provided musical score using the relative proportion methodology applied during training.

3.3 Prosody Encoder

Prosody affects how the content is expressed differently for speech and singing. In CookVoice, we categorize control signals of prosody, denoted as $\mathcal{P}$, into discrete and continuous signals. The expanded prosody token is denoted as $\mathcal{P}_e \in \mathbb{R}^{D\times T}$.

Discrete prosody signals come from either the G2P module (for speech) or musical scores (for singing voice). For speech, these consist of the aforementioned lexical prosody tokens P_{lex} (tones and stresses). Since the sequence length of P_{lex} matches the phonemes (L_1), each token has the same duration $\mathcal{D}^i$ of its corresponding phoneme. Consequently, the mapped token embeddings are repeated $\mathcal{D}^i$ times, giving the expanded discrete speech prosody embedding $P_{disc,e} \in \mathbb{R}^{D\times T}$.

For singing voice, the discrete signals are musical note tokens $P_{note} \in \mathbb{R}^{L_2}$. Similarly to the duration expansion in the Phoneme Encoder, we compute the exact frame duration for each note based on the score's beat allocation and the total frames T. The note embeddings are then expanded accordingly, serving as the singing counterpart for $P_{disc,e}$.

Continuous prosody signals are extracted directly from the reference voice waveform to explicitly represent fundamental frequency variations over time. We extract a continuous F_0 contour $\in \mathbb{R}^T$, which is passed through a linear projection layer to generate the continuous prosody embedding $P_{cont} \in \mathbb{R}^{D\times T}$.

During model training, the prosody embedding $\mathcal{P}_e \in \mathbb{R}^{D\times T}$ is constructed by randomly selecting the discrete representation ($P_{disc,e}$) or the continuous representation (P_{cont}) with equal probability. This strategy ensures that the model learns robust prosody control from both discrete tokens and continuous signals.

3.4 Flow-Matching DiT

Given the style S_e, content X_e, and prosody $\mathcal{P}_e$ satisfy $S_e, X_e, \mathcal{P}_e \in \mathbb{R}^{D\times T}$, we concatenate them along the feature dimension to construct a conditioning embedding $C \in \mathbb{R}^{3D\times T}$:

$$C = [S_e; X_e; \mathcal{P}_e]$$

In the final stage of CookVoice, our objective is to translate the condition embedding $C \in \mathbb{R}^{3D\times T}$ into high-fidelity latent embedding. To achieve this, we use a Diffusion Transformer (DiT) Peebles & Xie (2023) as the core generative architecture, where C serves as the essential cross-attention conditioning signal to guide the voice generation process.

We frame the generative process within the Optimal Transport Flow-Matching Lipman et al. (2022) (OT-FM) paradigm. Assuming a standard Gaussian prior $Y_0 \sim \mathcal{N}(0, I)$ and a target voice latent embedding $Y_1 \in \mathbb{R}^{N\times T}$, our DiT network, parameterized as $v_\theta(Y_t, t, C)$, is trained to regress the target vector field $Y_1 - Y_0$ that drives the linear probability path between the prior and the empirical

data distribution. The model is optimized using the standard Flow-Matching Mean Squared Error (MSE) objective.

$$\mathcal{L}_{FM} = \mathbb{E}_{t,Y_0,Y_1}\left[\|v_\theta(Y_t, t, C) - (Y_1 - Y_0)\|^2\right] \tag{1}$$

During inference, we sample the noise Y_0 and simulate the empirical probability flow Ordinary Differential Equation (ODE), given by $dY_t = v_\theta(Y_t, t, C)dt$. To solve this ODE and map the unstructured noise to the final acoustic representation Y_1, we employ a standard first-order Euler numerical solver. Starting from $t = 0$, the trajectory is discretely integrated with a fixed step size Δt to transform noise Y_0 into latent embedding condition on C.

$$Y_{t+\Delta t} = Y_t + \Delta t \cdot v_\theta(Y_t, t, C) \tag{2}$$

CookVoice supports multiple generation tasks through condition switching during training. For each sample, the style and prosody conditions are independently sampled from text-/voice-based style embeddings and discrete/continuous prosody. This allows a single model to learn different control combinations without task-specific heads or objectives. Detailed sampling formulations are provided in Appendix C.

4 Experiments

Dataset We combine multiple open-sourced bilingual (english and chinese) speech and singing voice datasets to conduct the experiment for broad coverage across diverse speakers, speaking styles, and singing styles. Selected datasets include *Baker* Databaker (2020), *LJSpeech* Ito & Johnson (2017), *ESD* Zhou et al. (2021), *CREMA-D* Cao et al. (2014), *CommonPhone* Klumpp et al. (2022), *Genshin Voice* dataset Simon3000 (2025), GTSinger Zhang et al. (2024c). The combined dataset contains approximately *123k* voice samples, *168* hours of voices from *6,361* speakers.

For continuous prosody control, we extract the F_0 contour from each voice sample and convert it into a normalized relative scale over each voiced frames. This reduces speaker-dependent absolute pitch information that may overlap with the style embedding. Detailed preprocessing is provided in Appendix D.

Implementation Details The CookVoice model is trained on **110k** samples, with a singing-to-speech ratio of 1:9. All evaluation samples are not seen by the model during training stage. For each sample, the temporally aligned style, content, and prosody embeddings are concatenated along the feature dimension to form the conditioning embedding. This conditioning embedding is then used as the condition for DiT-S backbone Peebles & Xie (2023), which learns to generate the target latent acoustic embedding. The model is optimized with the Flow-Matching objective Lipman et al. (2022). We train CookVoice on a single NVIDIA RTX 5090 GPU with 32 batch size for 800K steps.

Evaluation Metrics We evaluate CookVoice in terms of perceptual quality, intelligibility, style controllability, and prosody controllability. Subjective quality is measured using MOS, with M-CMOS additionally used for singing melody consistency. For objective evaluation, generated voices are transcribed using Whisper Radford et al. (2023), from which WER, PhoER, and ProER are computed. We further use S-SIM Lou et al. (2026a) for style similarity and F0-RMSE and F0-CORR for prosody fidelity. Detailed definitions and evaluation protocols are provided in Appendix E.

5 Results & Discussion

In this section, we provide an analysis of experimental results and organize the discussion to answer the following five research questions (**RQs**):

- **RQ1:** How does CookVoice perform compared with existing TTS and TTSV baselines in term of audio?
- **RQ2:** How controllable is CookVoice in terms of style and prosody compared with existing baselines?
- **RQ3:** How do different style and prosody control signals affect the generation performance?

Table 1: Comparison table of voice generation models. T/V denote text-/voice-based style control, and D/C denote discrete/continuous prosody control.

	Functional Coverage			Perceptual Quality		Controllability		Computational Efficiency		
Model	Task	Style	Prosody	TTS MOS	TTSV MOS / MC-MOS	S-SIM (%) TTS / TTSV	F0-CORR TTS / TTSV	#Para	CUDA Mem.	RTF
CosyVoice Du et al. (2024)	TTS	T+V	D	4.30	– / –	46.67 / –	0.1902 / –	436.00M	2.05G	3.74
F5-TTS Chen et al. (2025)	TTS	V	D	4.35	– / –	66.29 / –	0.1505 / –	300.00M	2.15G	6.15
ParaStyleTTS Lou et al. (2025a)	TTS	T	D	4.03	– / –	42.96 / –	0.1777 / –	52.51M	1.22G	**0.03**
IndexTTS Deng et al. (2025)	TTS	V	D	**4.42**	– / –	71.19 / –	0.2548 / –	608.00M	1.53G	0.36
Vevo2 Zhang et al. (2026)	TTS+TTSV	V	D+C	4.30	**3.42** / 0.11	75.11 / 88.09	0.1242 / 0.6242	872.00M	6.84G	14.85
DiffSinger Liu et al. (2022)	TTSV	T	D	–	2.80 / −0.80	– / 75.81	– / 0.4580	**26.74M**	**0.60G**	0.34
StyleSinger Zhang et al. (2024a)	TTSV	V	D	–	3.20 / −0.40	– / 82.00	– / 0.4856	42.00M	0.88G	0.36
TCSinger Zhang et al. (2024b)	TTSV	V	D	–	2.40 / −0.95	– / 87.53	– / 0.5643	329.50M	1.97G	0.33
Vevo1.5 Zhang et al. (2025)	TTSV	V	C	–	3.33 / −0.74	– / 86.55	– / 0.7125	1468.43M	9.44G	9.83
Vevo2 Zhang et al. (2026)	TTS+TTSV	V	D+C	4.30	**3.42** / 0.11	75.11 / 88.09	0.1242 / 0.6242	872.00M	6.84G	14.85
CookVoice	**TTS+TTSV**	**T+V**	**D+C**	3.98	3.40 / **0.28**	**91.65 / 95.00**	**0.7102 / 0.8425**	43.51M	1.37G	0.04

- **RQ4:** How does the number of ODE solver steps affect the quality-efficiency trade-off?
- **RQ5:** How efficient is CookVoice compared with existing TTS and TTSV models?

5.1 RQ1: Performance Comparison

We compare CookVoice with representative TTS and TTSV baselines in terms of controllability, perceptual quality, and intelligibility using both subjective and objective metrics. As shown in Tables 5 and 6, CookVoice achieves competitive generation quality while providing stronger style and prosody controllability across both TTS and TTSV tasks.

The main advantage of CookVoice lies in controllability. For TTS, CookVoice consistently improves style similarity and prosody consistency over the compared baselines. Under voice-based style control, CookVoice achieves an S-SIM of over 90% and an F0-CORR of up to 0.7102, whereas the best TTS baselines (Vevo2) achieve an SSIM of approximately 75% and an F0-CORR of approximately 0.25. A similar trend is observed for TTSV, CookVoice achieves the highest style similarity and prosody consistency, reaching an S-SIM of 95.00% and an F0-CORR of 0.8425. These results demonstrate that CookVoice more accurately preserves the style instruction from either voice or text modality and follows the prosody sourced from different types of control signals.

In terms of perceptual quality, CookVoice exhibits different performance patterns for speech and singing voice generation. For TTS, its MOS scores are lower than most TTS baselines, although its best configuration achieves a MOS of 3.98, close to the ground-truth score of 4.05. For TTSV, CookVoice achieves a best MOS of 3.40, outperforming most of the compared singing voice generation systems and remaining competitive to Vevo2, which achieves 3.42. CookVoice also obtains the highest MC-MOS score with better perceived melody consistency.

For intelligibility, CookVoice achieves competitive WER results. Its TTS performance lies in the middle range of the evaluated baselines. For TTSV, CookVoice outperforms most baselines on the Chinese song and achieves the second-lowest English WER among the models that support English singing voice generation. One possible reason for the generally higher WERs in TTSV is that Whisper is primarily trained on large-scale speech data. Singing voices exhibit different prosodic patterns, including wider pitch variations and prolonged phonemes, which may increase recognition errors. CookVoice achieves competitive performance with existing baseline who is trained on substantially larger datasets. Its primary advantages lie in its stronger controllability and flexible support for different style and prosody control signals.

5.2 RQ2: Controllability Analysis

As shown in Table 5, existing baselines typically support only a limited combination of style and prosody conditions. In contrast, CookVoice accommodates both text- and voice-based style control together with discrete prosody and continuous F_0 control. These control signals are explicitly aligned with the acoustic representation at the frame level, enabling finer-grained guidance throughout the generation process.

Compared with baselines under matched style–prosody settings, CookVoice delivers clear improvements. For TTS with text-based style and discrete prosody control, CookVoice improves S-SIM by 41.48% and F0-CORR by 121.10%, while reducing F0-RMSE by 5.53%. With voice-based style and discrete prosody control, CookVoice improves S-SIM by 16.46% and F0-CORR by 109.18%, while reducing F0-RMSE by 19.89%. These results suggest that CookVoice benefits from its explicit frame-level conditioning design. Outperforming existing voice cloning methods that rely primarily on autoregressive token prediction or voice-level style conditioning.

A similar trend is observed for TTSV. Under text-based style and discrete note control, CookVoice improves SSIM by 13.11% and F0-CORR by 17.53%, while reducing F0-RMSE by 2.74%. Under voice-based style and continuous F_0 control, it improves S-SIM by 7.84% and F0-CORR by 18.25%, while reducing F0-RMSE by 28.41%. The higher level of controllability of CookVoice can be attributed to the joint effect of its multimodal control interfaces and fine-grained frame-level conditioning. Together, these designs enable a broader range of controllable generation functions while improving both style and prosody controllability across speech and singing voice generation.

5.3 RQ3: Effects of Different Control Signals

In this section, we analyze how different control signals affect the performance of CookVoice. Figure 6 summarizes the marginal effects of different style and prosody control signals. Each value represents the average normalized performance of one control source across its matched performance, a higher value indicates better performance. The results reveal effects from the style source and the prosody source.

Effect of style control source. For TTS in Figure 6a, voice-based style conditioning consistently outperforms text-based conditioning across all evaluated metrics. The largest difference is observed in SSIM, where Voice achieves a normalized score of 0.932, compared with 0.199 for Text. Voice conditioning also produces better prosody consistency, perceptual quality, and intelligibility, achieving substantially higher scores for F0-CORR, MOS, and both Chinese and English WER. Voices provide richer acoustic information than natural language style prompts. It allows the model to preserve paralinguistic details such as timbre, emotion and age from the reference voice more accurately.

As shown in figure 6b. Voice conditioning achieves the highest SSIM and improves both MOS and MC-MOS, confirming its advantage for accurate singing-style transfer. However, its effect on intelligibility is less consistent. Voice and Text have similar Chinese WER performance, while Text achieves a substantially better English WER score. Although voice references improve style preservation and perceptual quality, stronger acoustic style imitation does not necessarily improve lyric intelligibility, particularly for English singing.

Effect of prosody control source. Continuous prosody signals F_0 provides a clear advantage over discrete prosody signals in prosody-related metrics. For TTS, CONT achieves scores of 0.844 for F0-RMSE and 0.892 for F0-CORR, compared with 0.230 and 0.221 for DISC. It also improves MOS and WER, indicating that detailed frame-level pitch guidance benefits not only prosody following but also overall speech quality and intelligibility.

The advantage of continuous control is even more pronounced for TTSV. CONT substantially outperforms DISC in F0-RMSE, F0-CORR, MOS, MC-MOS, and Chinese WER, confirming that explicit frame-level F_0 trajectories are effective for melody following and perceptual singing quality. However, unlike in TTS, DISC achieves higher S-SIM and better English WER performance in TTSV. One possible explanation is that the discrete prosody signal in TTSV consists of musical note, which already provide a structured representation of pitch, duration, and note boundaries that is closely aligned with the lyrics. This flexibility can help preserve the style and maintain clearer lyric articulation. Conversely, continuous signals contain detailed prosody trajectory from the reference voice. Which leads to an observable style-sim degradation under text-based style conditioning, where replacing discrete notes with continuous F_0 reduces SSIM from 85.75% to 72.51%.

5.4 RQ4: Effect of ODE Steps

We further analyze how performance is affected by the number of ODE solver steps, with detailed results provided in Appendix F. Increasing the number of ODE steps generally improves style simi-

larity and prosody fidelity, particularly under voice-based style conditioning and continuous prosody control.

Due to the high cost of collecting MOS and MC-MOS ratings, we evaluate these subjective metrics at coarser intervals, increasing the number of ODE steps by a factor of four, while the objective metrics are evaluated at each doubling of the step count. MOS continues to improve up to 16 steps before declining, whereas MC-MOS reaches its highest value at 4 steps and gradually decreases thereafter. Meanwhile, style similarity and F_0-based metrics generally stabilize between 4 and 8 steps, and intelligibility metrics such as WER, PhoER, and ProER achieve their best or near-best results at approximately 4 steps and may slightly degrade with further refinement. These results indicate that additional ODE steps can improve acoustic naturalness and prosodic details, but do not necessarily benefit melody consistency or linguistic content preservation.

Based on these observations, we suggest 4–8 ODE steps as the optimal operating range for CookVoice. Within this range, the generated voices maintain stable controllability, good perceptual quality, and intelligibility without incurring additional inference steps.

5.5 RQ5: Efficiency Analysis

Table 1 provides an integrated comparison of functional coverage, generation performance, controllability, and computational efficiency. The results reveal that existing systems typically optimize only a subset of these dimensions. Large-scale TTS models such as F5-TTS and IndexTTS achieve higher speech MOS scores than CookVoice, but are restricted to speech generation and require approximately $6.9\times$ and $14.0\times$ more parameters, respectively. Similarly, lightweight task-specific models such as ParaStyleTTS, DiffSinger, and StyleSinger provide competitive efficiency, but support only speech or singing voice generation and a limited set of control signals.

Vevo2 is the most directly competitive baseline because it supports both speech and singing voice generation. Vevo2 achieves higher TTS MOS and a marginally higher TTSV MOS than CookVoice. However, CookVoice obtains a higher MC-MOS and substantially stronger style and prosody controllability across both tasks. More importantly, CookVoice uses only 4.99% of the parameters, 20.03% of the CUDA memory, and 0.27% of the RTF required by Vevo2.

These comparisons indicate that the main advantage of CookVoice does not lie in maximizing a single task-specific quality metric. Instead, CookVoice provides a favorable trade-off among functional coverage, fine-grained controllability, perceptual quality, and computational efficiency. With a single 43.51M-parameter model, it supports both TTS and TTSV, text- and voice-based style conditioning, and discrete and continuous prosody control, while maintaining real-time generation with an RTF of 0.04.

The compact model size and fast inference of CookVoice mainly stem from its streamlined architecture. By replacing autoregressive token prediction with a lightweight DiT-S generative backbone, CookVoice limits the model size to 43.51M parameters and avoids the sequential next-token decoding required by AR models. In addition, CookVoice employs flow matching, which requires only a small number of ODE solver steps, compared with conventional diffusion sampling processes that typically involve hundreds of iterative denoising steps Ho et al. (2020). These design choices reduce both model complexity and inference cost.

6 Conclusion

In this paper, we present **CookVoice**, a unified framework for multimodal, multi-style, and multi-task human voice generation. By decomposing human voice into content, prosody, and style, CookVoice supports a wide range of speech and singing voice generation tasks within a single model, including TTS, TTSV, style-controllable generation, voice mimicry, and voice conversion. To enable fine-grained controllability, we design a flexible alignment strategy that expands different control signals to the level of the acoustic frame, allowing text, reference voice, discrete prosody, and continuous F_0 contours to be combined in a unified generation process. Experimental results show that CookVoice achieves competitive generation quality to existing task-specific baselines while providing stronger style and prosody controllability, smaller model size and efficient inference speed.

7 GENERATIVE AI DISCLOSURE

In this work, we used generative AI tools only to polish and improve the clarity of the manuscript writing. We did not use generative AI tools for code writing, experimental design, experiment execution, result analysis, or evaluation. All technical implementation, experimental design, experiments, and evaluations were conducted by the authors. All AI-assisted text was carefully reviewed and revised by the authors to ensure technical accuracy and consistency with the original research. We take responsibility for the final content of this work, including any text produced with the aid of generative AI.

REFERENCES

Houwei Cao, David G Cooper, Michael K Keutmann, Ruben C Gur, Ani Nenkova, and Ragini Verma. Crema-d: Crowd-sourced emotional multimodal actors dataset. *IEEE transactions on affective computing*, 5(4):377–390, 2014.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, 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.

Databaker. Chinese mandarin female corpus. `https://en.data-baker.com/datasets/freeDatasets/`, 2020. Accessed: 2023-04-20.

Wei Deng, Siyi Zhou, Jingchen Shu, Jinchao Wang, and Lu Wang. Indextts: An industrial-level controllable and efficient zero-shot text-to-speech system. *arXiv preprint arXiv:2502.05512*, 2025.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Keith Ito and Linda Johnson. The lj speech dataset. `https://keithito.com/LJ-Speech-Dataset/`, 2017.

Jaehyeon Kim, Jungil Kong, and Juhee Son. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International conference on machine learning*, pp. 5530–5540. PMLR, 2021.

Philipp Klumpp, Tomas Arias, Paula Andrea Pérez-Toro, Elmar Noeth, and Juan Orozco-Arroyave. Common phone: A multilingual dataset for robust acoustic modelling. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 763–768, 2022.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033, 2020.

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

Jinglin Liu, Chengxi Li, Yi Ren, Feiyang Chen, and Zhou Zhao. Diffsinger: Singing voice synthesis via shallow diffusion mechanism. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 11020–11028, 2022.

Haowei Lou, Hye-Young Paik, Wen Hu, and Lina Yao. Parastyletts: Toward efficient and robust paralinguistic style control for expressive text-to-speech generation. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 1979–1988, 2025a.

Haowei Lou, Hye-young Paik, Sheng Li, Wen Hu, and Lina Yao. Generalized multilingual text-to-speech generation with language-aware style adaptation. *arXiv preprint arXiv:2504.08274*, 2025b.

Haowei Lou, Junda Wu, Chengkai Huang, Tong Yu, Hye-young Paik, Wen Hu, and Lina Yao. Autosift: Automatic style sifting for controllable speech generation with arbitrary style infilling. *arXiv preprint arXiv:2607.12706*, 2026a.

Haowei Lou, Hye young Paik, Wen Hu, and Lina Yao. Parameta: Towards learning disentangled paralinguistic speaking styles representations from speech. In *AAAI Conference on Artificial Intelligence*, 2026b. URL https://api.semanticscholar.org/CorpusID:284910281.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

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.

Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech: Fast, robust and controllable text to speech. *Advances in neural information processing systems*, 32, 2019.

Simon3000. Genshin voice: A multi-lingual voice dataset from Genshin Impact. https://huggingface.co/datasets/simon3000/genshin-voice, 2025. Hugging Face Datasets.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33: 16857–16867, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al. Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135*, 2017.

Lichao Zhang, Ruiqi Li, Shoutong Wang, Liqun Deng, Jinglin Liu, Yi Ren, Jinzheng He, Rongjie Huang, Jieming Zhu, Xiao Chen, et al. M4singer: A multi-style, multi-singer and musical score provided mandarin singing corpus. *Advances in Neural Information Processing Systems*, 35: 6914–6926, 2022.

Xueyao Zhang, Xiaohui Zhang, Kainan Peng, Zhenyu Tang, Vimal Manohar, Yingru Liu, Jeff Hwang, Dangna Li, Yuhao Wang, Julian Chan, Yuan Huang, Zhizheng Wu, and Mingbo Ma. Vevo: Controllable zero-shot voice imitation with self-supervised disentanglement. In *ICLR*. OpenReview.net, 2025.

Xueyao Zhang, Junan Zhang, Yuancheng Wang, Chaoren Wang, Yuanzhe Chen, Dongya Jia, Zhuo Chen, and Zhizheng Wu. Vevo2: A unified and controllable framework for speech and singing voice generation. *IEEE Transactions on Audio, Speech and Language Processing*, 2026.

Yu Zhang, Rongjie Huang, Ruiqi Li, JinZheng He, Yan Xia, Feiyang Chen, Xinyu Duan, Baoxing Huai, and Zhou Zhao. Stylesinger: Style transfer for out-of-domain singing voice synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19597–19605, 2024a.

Yu Zhang, Ziyue Jiang, Ruiqi Li, Changhao Pan, Jinzheng He, Rongjie Huang, Chuxin Wang, and Zhou Zhao. Tcsinger: Zero-shot singing voice synthesis with style transfer and multi-level style control. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1960–1975, 2024b.

Yu Zhang, Changhao Pan, Wenxiang Guo, Ruiqi Li, Zhiyuan Zhu, Jialei Wang, Wenhao Xu, Jingyu Lu, Zhiqing Hong, Chuxin Wang, et al. Gtsinger: A global multi-technique singing corpus with realistic music scores for all singing tasks. *Advances in Neural Information Processing Systems*, 37:1117–1140, 2024c.

Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 920–924. IEEE, 2021.

Supplementary Material

A Related Work

Autoregressive voice generation. Recent advances in neural voice generation have significantly improved the naturalness and expressiveness of synthesized speech. Representative systems such as CosyVoice Du et al. (2024) and IndexTTS Deng et al. (2025) adopt autoregressive (AR) generation mechanisms to model acoustic token sequences. Using token-by-token prediction and large-scale training, these models can generate expressive high-quality speech. However, AR decoding also introduces several limitations for controllability. Since duration, rhythm, and prosody are implicitly determined by the generated token sequence, the generation process is difficult to interpret and hard to control at a fine temporal granularity. This becomes especially problematic when the target task requires explicit duration or pitch control, such as singing voice generation, prosody mimicry, and frame-level voice editing. In addition, AR generation usually suffers from slower inference due to its sequential decoding process.

NAR and controllable speech generation. NAR methods improve inference efficiency and temporal stability by introducing explicit or semi-explicit alignment mechanisms. For example, FastSpeech-style models rely on duration prediction and phoneme-level expansion to generate speech in parallel, while recent flow-matching or diffusion-based systems further improve acoustic quality through continuous generative modeling Chen et al. (2025). These methods provide better control over phoneme-level timing than AR models. However, most existing NAR speech generation systems are still designed for specific tasks, such as TTS, expressive TTS, or voice cloning. Their control interfaces are often limited to either text prompts, speaker references, or phoneme-level durations, making it difficult to flexibly combine content, style, prosody, and singing-related controls under a unified framework.

Singing voice generation. Singing voice generation requires more explicit temporal and melodic control than ordinary speech synthesis. Traditional singing voice synthesis systems, such as DiffSinger Liu et al. (2022), usually rely on musical scores and note-level conditions to generate singing voices. More recent work introduces style control into singing voice generation. For example, StyleSinger Zhang et al. (2024a) improves controllability by incorporating singing style information. However, such systems often assume a predefined relationship between lyrics, phonemes, and notes. In particular, binding notes directly with phonemes simplifies alignment, but it may not always reflect the flexible temporal structure of real singing, where one phoneme may span multiple notes or one note may correspond to multiple phonetic units. This type of restricted alignment improves tractability but limits flexible control in more general singing voice generation scenarios.

Unified voice generation. Recent unified voice generation models attempt to cover multiple voice generation tasks using a single model. For example, Vevo-style systems Zhang et al. (2025; 2026) move toward unified speech and singing voice modeling. However, many of these systems still rely on AR decoding or use global reference conditions, such as utterance-level or sentence-level prosody references. While these designs are effective for general voice generation, they provide limited direct control over fine-grained temporal details. In particular, when content, style, notes, lexical tones, stresses, and continuous F_0 contours are jointly involved, existing unified models still struggle to provide flexible frame-level controllability.

B Latent Acoustic Encoding

We employ an Autoencoder (AE) with similar architecture with the HiFi-GAN Kong et al. (2020) to efficiently encode audio signals into latent embedding. The encoder takes a linear spectrogram $\mathrm{spec} \in \mathbb{R}^{513\times T}$ as input and compresses it into a continuous latent embedding $Y \in \mathbb{R}^{N\times T}$, where N denotes the latent channel dimension and T represents the temporal frame length. The decoder is constructed using a series of transposed 1D convolutional layers, which progressively upsample the latent embedding Y to reconstruct the original audio waveform. To ensure high-fidelity audio generation, the AE is jointly optimized using an adversarial loss from a discriminator alongside a reconstruction loss.

C MULTI-TASK TRAINING

CookVoice supports multiple generation tasks through a simple condition-switching training strategy. Instead of designing task-specific objectives or separate task heads, we randomly replace the conditioning embeddings during training. For each sample in the training batch, the style condition is randomly sampled from either the text-based style embedding $S_{\mathcal{T}}$ or the voice-based style embedding $S_{\mathcal{V}}$. Similarly, the prosody condition is randomly sampled from either the discrete prosody embedding $\mathcal{P}_{disc,e}$ or the continuous F_0-based prosody embedding $\mathcal{P}_{cont}$:

$$S = \begin{cases} S_{\mathcal{T}}, & p = 0.5, \\ S_{\mathcal{V}}, & p = 0.5, \end{cases} \quad \mathcal{P}_e = \begin{cases} \mathcal{P}_{disc,e}, & p = 0.5, \\ \mathcal{P}_{cont}, & p = 0.5. \end{cases} \tag{3}$$

This randomization is applied at the sample level within the batch, so different samples may use different style and prosody sources in the same training step. As a result, CookVoice learns to generate voices under different combinations of control signals, such as text style with discrete prosody, voice style with discrete prosody, text style with continuous F_0, and voice style with continuous F_0. This strategy allows a single model to support multiple speech and singing voice generation tasks without changing the model architecture or training objective.

D PREPROCESSING

We extract the raw F_0 contour (in Hz) from each voice sample. To accommodate the wide and varying pitch ranges inherent to both speech and singing voices, we project the F_0 values onto a logarithmic scale and normalize them using a lower of 50 Hz.

$$\tilde{f} = \frac{\log \mathrm{F}_0 - \log f_{\min}}{\log f_{\max} - \log f_{\min}} \quad f_{\min} = 50, \quad \hat{f} = \begin{cases} \tilde{f}_t - \mathrm{mean}(\tilde{f}_t) & t \in \mathcal{U} \\ -1, & t \notin \mathcal{U} \end{cases} \tag{4}$$

$\tilde{f}_t$ provides a standardized absolute pitch, relying on it directly can introduce entanglement. Paralinguistic styles, such as age and gender, can effectively dictate a speaker's absolute pitch range and are already captured by the style embedding. Absolute F_0 as an explicit condition would therefore cause interference between prosody and style. To decouple the absolute pitch from the speaker's timbre, we convert the absolute F_0 into a relative pitch contour by removing its voice-level mean. $\mathcal{U}$ denotes voiced frames. For unvoiced frames, we set a constant value of -1 to allow the relative $\hat{f}$ to capture pitch changes over time without introducing absolute pitch information that is already captured in the style embedding.

E EVALUATION METRICS

This section provides detailed definitions of the evaluation metrics used in our experiments.

Word Error Rate (WER). WER measures word-level transcription errors between the generated audio and the ground-truth text. We first transcribe the generated audio using Whisper Radford et al. (2023), and then compute the edit distance between the predicted word sequence and the reference word sequence:

$$\mathrm{WER} = \frac{S + D + I}{N}, \tag{5}$$

where S, D, and I denote the number of substitutions, deletions, and insertions, respectively, and N is the number of words in the reference text. Lower WER has better intelligibility.

Phoneme Error Rate (PhoER). PhoER measures content accuracy at the phoneme level. We convert both the recognized text and the ground-truth text into phoneme sequences using the same G2P module, and then compute the normalized edit distance:

$$\mathrm{PhoER} = \frac{S_p + D_p + I_p}{N_p}, \tag{6}$$

where S_p, D_p, and I_p denote phoneme-level substitutions, deletions, and insertions, and N_p is the number of phonemes in the reference sequence. A lower PhoER indicates more accurate pronunciation and content preservation.

Prosody Error Rate (ProER). ProER measures the error rate of discrete prosody tokens, such as Mandarin tones, English stresses, or note-related prosody labels. Similar to PhoER, we compute the normalized edit distance between the predicted and reference prosody-token sequences:

$$\mathrm{ProER} = \frac{S_r + D_r + I_r}{N_r}, \tag{7}$$

where S_r, D_r, and I_r denote substitution, deletion, and insertion errors in the prosody-token sequence, and N_r is the number of reference prosody tokens. A lower ProER indicates better preservation of discrete prosodic information.

Style Similarity (S-SIM). S-SIM evaluates whether the generated voice matches the target style. We extract style embeddings from the generated audio and the target reference using a pre-trained style encoder. The cosine similarity between the two embeddings is computed as:

$$\text{S-SIM} = \frac{\mathbf{s}_{gen}^{\top}\mathbf{s}_{ref}}{\|\mathbf{s}_{gen}\|_2\|\mathbf{s}_{ref}\|_2}, \tag{8}$$

where $\mathbf{s}_{gen}$ and $\mathbf{s}_{ref}$ are the style embeddings of the generated and reference voices, respectively. A higher S-SIM indicates better style preservation.

F0 Root Mean Square Error (F0-RMSE). F0-RMSE measures the absolute pitch difference between the generated and target F0 contours. We compute F0-RMSE over voiced frames where both generated and reference F0 values are valid:

$$\text{F0-RMSE} = \sqrt{\frac{1}{T_v}\sum_{t=1}^{T_v}\left(f_t^{gen} - f_t^{ref}\right)^2}, \tag{9}$$

where f_t^{gen} and f_t^{ref} are the generated and reference F0 values at frame t, and T_v is the number of valid voiced frames. A lower F0-RMSE indicates smaller absolute pitch deviation.

F0 Pearson Correlation (F0-CORR). F0-CORR measures whether the generated pitch contour follows the same trend as the reference contour:

$$\text{F0-CORR} = \frac{\sum_{t=1}^{T_v}(f_t^{gen} - \bar{f}^{gen})(f_t^{ref} - \bar{f}^{ref})}{\sqrt{\sum_{t=1}^{T_v}(f_t^{gen} - \bar{f}^{gen})^2}\sqrt{\sum_{t=1}^{T_v}(f_t^{ref} - \bar{f}^{ref})^2}}, \tag{10}$$

where $\bar{f}^{gen}$ and $\bar{f}^{ref}$ are the mean F0 values of the generated and reference contours. A higher F0-CORR indicates better preservation of pitch movement and prosodic trend.

Mean Opinion Score (MOS). MOS is used to evaluate perceptual naturalness and overall audio quality. Human listeners rate each generated sample on a scale from 1 to 5, where 1 indicates poor quality and 5 indicates excellent quality. The final MOS is the average score across all listeners and samples.

Melody Comparative MOS (M-CMOS). M-CMOS is used for singing voice evaluation. Listeners compare generated singing samples and judge which one better follows the target melody. This metric evaluates melody consistency from human perception, complementing objective F0-based metrics.

F Signal Impact Analysis

We analyze the impact of the number of ODE solver steps under different sources of style and prosody control. Specifically, we compare text-based and voice-based style control, as well as discrete and continuous prosody control, on both TTS and TTSV tasks. We evaluate three aspects of generation quality: style similarity, prosody fidelity, and intelligibility.

F.1 Style similarity

As shown in Fig. 2, style similarity generally increases as the number of ODE steps increases, and gradually converges around 4–8 steps. This suggests that most style-related information is formed

in the early part of the generation trajectory, while further increasing the number of steps brings diminishing returns. We also observe that reference-voice-based style control consistently outperforms text-based style control. This is expected because reference voices contain richer speaker identity and paralinguistic information than text style descriptions. In addition, continuous prosody control generally achieves higher style similarity than discrete prosody tokens, indicating that dense frame-level control signals provide more detailed guidance during generation.

F.2 Prosody fidelity

For prosody, we evaluate F0-RMSE and F0-Correlation, as shown in Fig. 3. F0-RMSE measures the absolute pitch deviation from the ground-truth contour, while F0-Correlation measures whether the generated pitch follows the overall trend of the target F0 trajectory. For F0-RMSE, the error generally decreases as the number of ODE steps increases, with voice-based style and continuous prosody control achieving better performance. This is also reasonable because the reference voice directly provides acoustic and pitch-related cues, while continuous F0 signals preserve more fine-grained pitch information than discrete tokens.

A similar trend can be observed for F0-Correlation, where the correlation improves with more ODE steps and converges around 8 steps. However, the improvement for discrete prosody control is relatively limited, especially in the TTSV setting. One possible reason is that singing voice often follows a more constrained and stable melodic structure compared with speech. Therefore, discrete prosody tokens may already capture the coarse pitch movement, and increasing the number of ODE steps mainly improves acoustic details rather than substantially changing the global F0 correlation.

F.3 Intelligibility

The intelligibility results are shown in Fig. 4. Different from style similarity and prosody fidelity, WER, PhoER, and ProER first decrease and reach their best performance at 4 ODE steps, but tend to increase when more ODE steps are used. This indicates that using too many ODE steps may over-refine acoustic, style, or prosodic details at the cost of linguistic clarity. In other words, a larger number of ODE steps does not always lead to better content preservation.

We also observe that TTSV generally has higher error rates than TTS. This may be partly attributed to the evaluation model, since Whisper Radford et al. (2023) is mainly trained and optimized for speech recognition rather than singing voice recognition. Moreover, singing voice naturally contains longer phoneme durations, stronger pitch variation, and melody-driven pronunciation changes, which can make recognition more difficult.

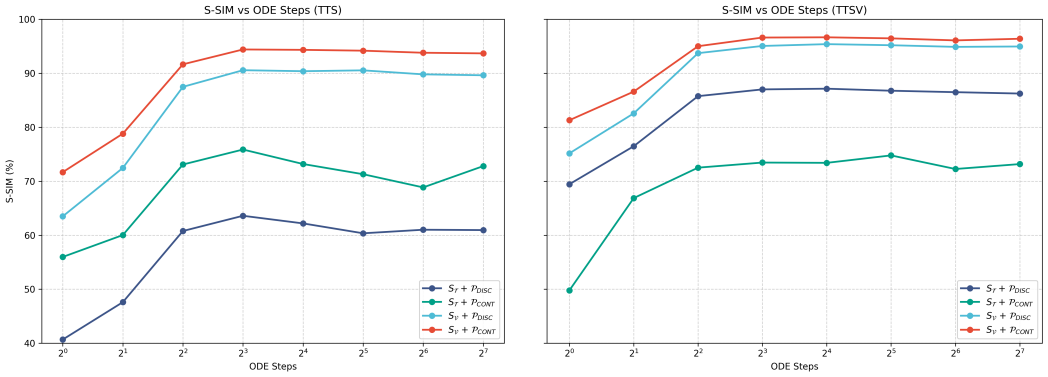
Overall, these results suggest that different generation aspects prefer different numbers of ODE steps. Style similarity and prosody fidelity benefit from increasing ODE steps and usually converge around 4–8 steps, while intelligibility achieves the best trade-off at around 4 steps. Therefore, a small number of ODE steps provides a better balance between controllability, acoustic quality, and content preservation.

G Limitation

CookVoice shows potential for unified and controllable human voice generation, several limitations remain. First, due to resource constraints, CookVoice has not yet been scaled up. The current model only use the DiT-S version of diffusion transformer (43.51M parameters) and is trained on 168 hours of data, which is still much smaller than large-scale voice generation systems. The scaling potential of CookVoice has not been fully explored. Another limitation lies in CookVoice mainly focuses on human voice generation. Its potential applicability to broader audio generation domains, such as music, instrumental sound, and general audio generation, has not yet been investigated.

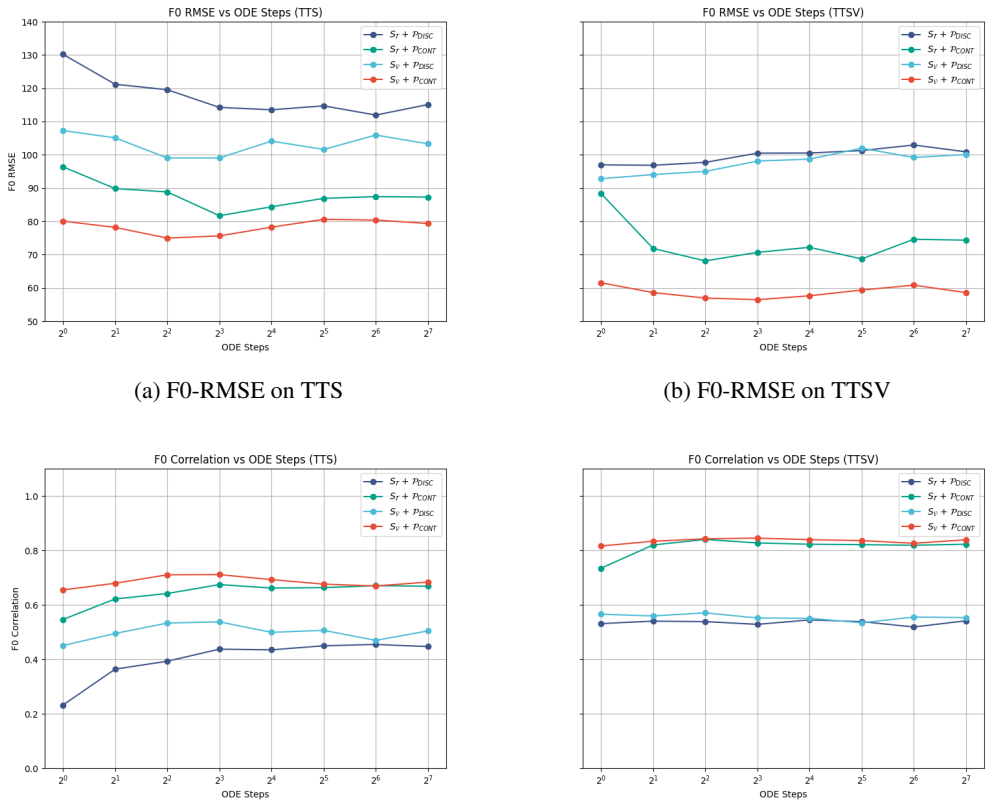
H Task Definition

CookVoice is designed as a unified framework for speech and singing voice generation. Instead of building separate models for different tasks, we formulate a broad range of generation, conversion,



(a) TTS (b) TTSV

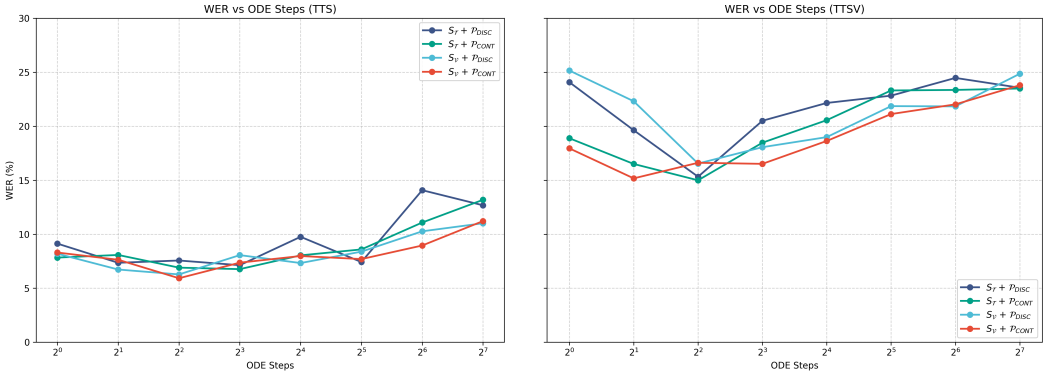
Figure 2: Style similarity under different numbers of ODE steps for TTS and TTSV.



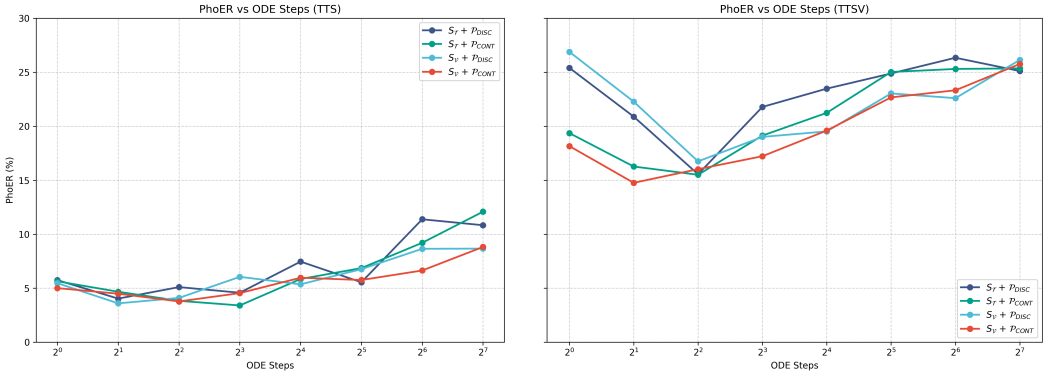
(a) F0-RMSE on TTS (b) F0-RMSE on TTSV

(c) F0-Correlation on TTS (d) F0-Correlation on TTSV

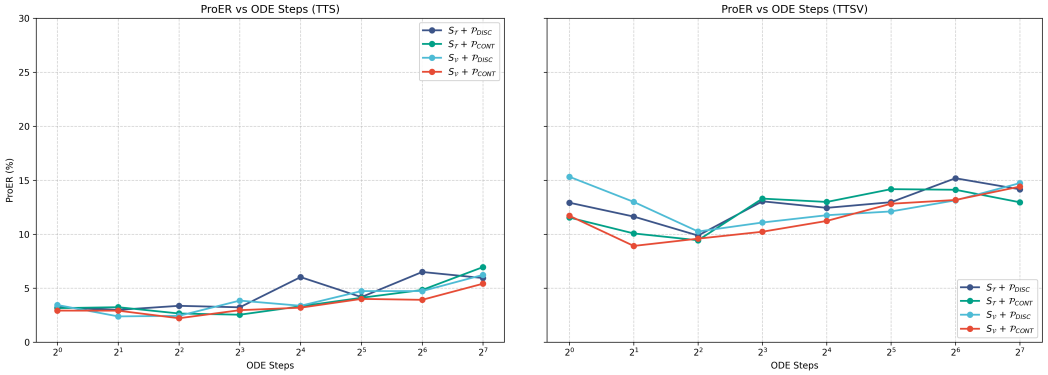
Figure 3: Prosody fidelity under different numbers of ODE steps. F0-RMSE measures absolute pitch deviation, while F0-Correlation measures the consistency of the generated F0 contour with the target contour.



(a) WER on TTS (b) WER on TTSV

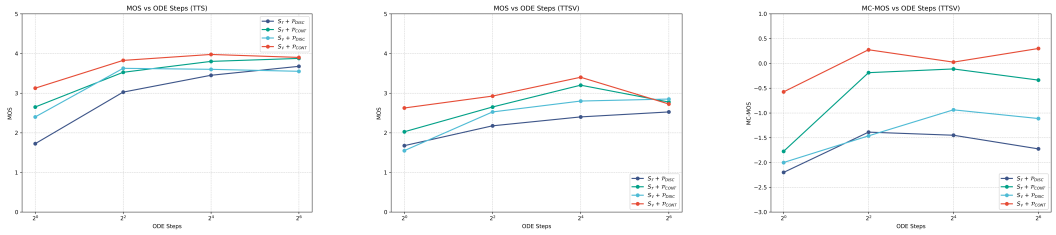


(c) PhoER on TTS (d) PhoER on TTSV



(e) ProER on TTS (f) ProER on TTSV

Figure 4: Intelligibility under different numbers of ODE steps for TTS and TTSV. WER, PhoER, and ProER are used to evaluate word-level, phoneme-level, and prosody-token-level error rates, respectively.



(a) MOS on TTS (b) MOS on TTSV (c) MC-MOS on TTSV

Figure 5: Human subjective listening test evaluation under different numbers of ODE steps. MOS measures overall naturalness, MC-MOS measures the melody accuracy of the generated singing voice.

Table 2: Alignment method and controllability comparison.

Model Class	**Alignment Method**	**Controllability**
AR	Token prediction	Voice-level
NAR	Restricted alignment	Phoneme-level
CookVoice	Flexible alignment	Frame-level

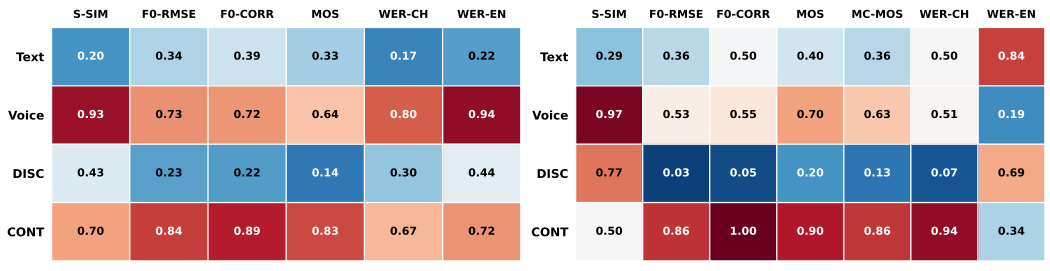
editing, and mimicry tasks under a shared conditional generation function $\mathcal{G}(\cdot)$. The key difference between tasks lies in the combination of control signals, including linguistic content, style conditions, lexical or musical prosody, and frame-level pitch contours.

Specifically, X represents the phoneme derived from text or lyric to be generated, while $S_{\mathcal{T}}$ and $S_{\mathcal{V}}$ denote style information provided by text descriptions and reference voices, respectively. Prosodic control can be provided at different granularities: $\mathcal{P}_{lex}$ represents discrete lexical prosody such as tones or stresses, $\mathcal{P}_{note}$ represents musical note-level conditions for singing voice generation, and F_0 provides continuous frame-level pitch control. By flexibly combining these conditions, CookVoice can support multiple tasks within a single model, as summarized in Table 3.

I Terminology

To make the paper accessible to readers without a background in linguistics, speech processing, or music, we summarize the key terms used in CookVoice in Table 4. These terms cover linguistic content, prosody, music-related control signals, style control, and evaluation metrics.

J Supplementary Results



(a) TTS (b) TTSV

Figure 6: Marginal effects of different style and prosody control signals on CookVoice's performance. Redder indicate stronger performance, bluer indicate weaker performance.

Table 3: Summary of generation tasks and control settings supported by CookVoice. X denotes phoneme/lyric content, $S_{\mathcal{T}}$ and $S_{\mathcal{V}}$ denote text-based and voice-based style conditions, $\mathcal{P}_{lex}$ denotes lexical prosody such as tones or stresses, $\mathcal{P}_{note}$ denotes musical note conditions, and F_0 denotes F0 contour.

Task Name	**Abbr.**	**Formulation**	**Description**
Text-to-Speech	TTS	$\mathcal{G}(X, \mathcal{P}_{lex})$	Generate a speech waveform from text content and lexical prosody.
Text Style-Controllable TTS	T-SC-TTS	$\mathcal{G}(S_{\mathcal{T}}, X, \mathcal{P}_{lex})$	Generate speech conditioned on a natural language style description, such as emotion, age, gender, or speaking manner.
Voice Style-Controllable TTS	V-SC-TTS	$\mathcal{G}(S_{\mathcal{V}}, X, \mathcal{P}_{lex})$	Generate speech that preserves the text content while mimicking the global style or timbre of a reference voice.
Prosody-Controllable TTS	PC-TTS	$\mathcal{G}(S, X, F_0)$	Generate speech with explicit frame-level prosody control from a reference F_0 contour while keeping the target content and style.
Text-to-Singing Voice	TTSV	$\mathcal{G}(X, \mathcal{P}_{note})$	Generate a singing voice from lyrics and musical notes.
Text Style-Controllable TTSV	T-SC-TTSV	$\mathcal{G}(S_{\mathcal{T}}, X, \mathcal{P}_{note})$	Generate singing voice from lyrics and notes while controlling singing style using a text prompt.
Voice Style-Controllable TTSV	V-SC-TTSV	$\mathcal{G}(S_{\mathcal{V}}, X, \mathcal{P}_{note})$	Generate singing voice from lyrics and notes while transferring the style or timbre from a reference voice.
Prosody-Controllable TTSV	PC-TTSV	$\mathcal{G}(S, X, \mathcal{P}_{F_0})$	Generate singing voice by following a continuous frame-level F_0 contour, enabling fine-grained melody or prosody mimicry.
Voice Mimicry	VM	$\mathcal{G}(S_{\mathcal{V}}, X, \mathcal{P}_{F_0})$	Generate speech with new content while jointly mimicking the reference voice style and detailed prosodic trajectory.
Voice Conversion	VC	$\mathcal{G}(S_{\mathcal{V}}^{tar}, X^{src}, \mathcal{P}_{F_0}^{src})$	Convert a source speech into a target voice style while preserving the source linguistic content and prosody.
Singing Voice Conversion	SVC	$\mathcal{G}(S_{\mathcal{V}}^{tar}, X^{src}, \mathcal{P}_{F_0}^{src})$	Convert a source singing voice into a target singing style while preserving lyrics and melodic contour.
Speech Editing	SE	$\mathcal{G}(S, X^{edit}, \mathcal{P}_{F_0})$	Edit the linguistic content or style of speech while maintaining the remaining prosodic structure.
Singing Voice Editing	SVE	$\mathcal{G}(S, X^{edit}, \mathcal{P}_{note/F_0})$	Edit lyrics, style, or melody-related controls in singing voice generation.
Sketch-to-Voice	STV	$\mathcal{G}(S, X, \mathcal{P}_{F_0}^{sketch})$	Generate speech or singing voice from a user-specified coarse pitch sketch or manually controlled F_0 trajectory.
Humming-To-Voice	HTV	$\mathcal{G}(S, X, \mathcal{P}_{F_0}^{humm})$	Generate speech or singing voice from a user-specified humming F_0 contour.

Table 4: Terminology used in CookVoice. The table provides simplified explanations of linguistic, musical, control, and evaluation terms for non-expert readers.

Term	Category	Meaning in CookVoice
Human Voice Generation (HVG)	General task	A broad task family that generates human speech or singing voice from control signals such as text, lyrics, notes, style prompts, reference voices, or pitch contours.
Text-to-Speech (TTS)	Speech task	Generate a spoken voice waveform from text or phoneme content.
Text-to-Singing Voice (TTSV)	Singing task	Generate a singing voice waveform from lyrics and musical note information.
Voice Conversion (VC)	Conversion task	Convert a source speech voice into a target speaker or style while preserving the source content and prosody.
Singing Voice Conversion (SVC)	Conversion task	Convert a source singing voice into a target singing voice style while preserving lyrics and melody.
Phoneme	Linguistics	The basic sound unit of speech. For example, a word is converted into a sequence of phonemes before being generated as speech or singing voice.
Grapheme-to-Phoneme (G2P)	Linguistics	A text processing module that converts written text or lyrics into phoneme sequences.
Lyrics	Music / linguistics	The textual content of a song. In CookVoice, lyrics are converted into phonemes and used as singing content.
Lexical prosody	Linguistics	Discrete pronunciation-related information attached to phonemes or words, such as Mandarin tones or English word stresses.
Tone	Linguistics	A pitch pattern that changes word meaning in tonal languages such as Mandarin Chinese. In CookVoice, tones are used as discrete prosody tokens for speech.
Stress	Linguistics	The emphasis placed on a syllable or word, often affecting loudness, duration, and pitch in English speech.
Prosody	Speech / music	The expressive pattern of voice, including pitch, rhythm, duration, stress, and intonation. Prosody controls how the content is spoken or sung.
Discrete prosody	Control signal	Symbolic prosody information such as tones, stresses, or MIDI notes. It provides coarse but practical control without requiring reference audio.
Continuous prosody	Control signal	Frame-level acoustic prosody information, mainly represented by the continuous F_0 contour. It provides fine-grained pitch control.
F_0	Acoustics	Fundamental frequency, corresponding to the perceived pitch of the voice. Higher F_0 generally means a higher-pitched voice.
F_0 contour	Acoustics	A time-varying sequence of F_0 values that describes how pitch changes over time. CookVoice uses it for frame-level continuous prosody control.
Waveform	Signal processing	The raw audio signal represented as amplitude values over time. It is the final audio output that listeners hear.
Spectrogram	Signal processing	A time-frequency representation of audio. It shows how the frequency content of a waveform changes over time, and is commonly used as an intermediate acoustic representation for voice generation.
Frame	Signal processing	The smallest unit of slice in the spectrogram along temporal axis. A voice waveform is converted into a spectrogram with multiple frames along the temporal axis, and CookVoice aligns content, style, and prosody to these acoustic frames for fine-grained control.
Latent acoustic embedding	Signal processing	A compressed acoustic feature extracted from the spectrogram. CookVoice generates the target latent acoustic embedding, which is then converted back into a waveform by the decoder.
Frame-level alignment	Model design	A mechanism that expands different control signals to match the acoustic frame length, allowing content, style, and prosody to be combined at each time step.
Duration	Timing	The length of time a phoneme or note lasts. Duration determines how content is expanded to match the acoustic frames.

Table 4: Terminology used in CookVoice. The table provides simplified explanations of linguistic, musical, control, and evaluation terms for non-expert readers. Continued.

Term	Category	Meaning in CookVoice
MIDI note	Music	A symbolic musical pitch representation used in singing voice generation. It specifies which note should be sung.
Note duration	Music	The length of a musical note, usually derived from the score or beat information. It controls how long a note is sustained in singing.
Beat	Music	A unit of musical timing. In CookVoice, beat information is used to compute relative note and phoneme durations for singing voice generation.
Melody	Music	The ordered pitch movement of singing. It is mainly controlled by musical notes or continuous F_0 contours.
Humming	Music / control	A user-provided vocal melody without full lyrics. CookVoice can use its extracted F_0 contour as a prosody control signal.
Pitch sketch	User control	A manually specified or coarse pitch trajectory used to guide the generated speech or singing voice.
Style	Voice attribute	Global paralinguistic characteristics of a voice, such as speaker identity, timbre, emotion, age, gender, and speaking or singing manner.
Timbre	Voice attribute	The tone color or acoustic identity of a voice. Two voices can sing the same note but sound different because of different timbres.
Text style prompt	Style control	A natural language description of the desired voice style, such as "a female speaking happily".
Reference voice	Style control	An example speech or singing voice used to provide target voice style, timbre, or speaker identity.
Text-based style	Style control	Style information extracted from a natural language prompt. It provides semantic-level style control.
Voice-based style	Style control	Style information extracted from a reference voice. It provides more direct acoustic style and timbre control.
WER	Evaluation metric	Word Error Rate. It measures word-level content errors after transcribing generated audio with an ASR system. Lower is better.
PhoER	Evaluation metric	Phoneme Error Rate. It measures phoneme-level pronunciation or content errors. Lower is better.
ProER	Evaluation metric	Prosody Error Rate. It measures errors in discrete prosody tokens such as tones and stresses. Lower is better.
S-SIM	Evaluation metric	Style Similarity. It measures how similar the generated voice style is to the target style. Higher is better.
F0-RMSE	Evaluation metric	Root mean square error between generated and target F_0 contours. It measures absolute pitch deviation. Lower is better.
F0-CORR	Evaluation metric	Pearson correlation between generated and target F_0 contours. It measures whether the pitch trend is preserved. Higher is better.
MOS	Subjective metric	Mean Opinion Score. Human listeners rate the naturalness or quality of generated audio, usually on a 1–5 scale.
M-CMOS	Subjective metric	Melody Comparative MOS. Human listeners compare whether one singing sample follows the melody better than another.

Table 5: Controllability comparison using S-SIM, F0-RMSE, and F0-CORR. Parentheses denote relative improvements over the best baseline.

Task	Model	Style + Prosody	SSIM (↑) %	F0-RMSE (↓)	F0-CORR (↑)
TTS	CosyVoice Du et al. (2024)	Text + DISC	30.53 ± 10.98	138.18 ± 43.17	0.1065 ± 0.1709
	CosyVoice Du et al. (2024)	Voice + DISC	46.67 ± 17.10	126.35 ± 45.23	0.1902 ± 0.2423
	F5-TTS Chen et al. (2025)	Voice + DISC	66.29 ± 13.73	136.56 ± 46.28	0.1505 ± 0.1703
	ParaStyleTTS Lou et al. (2025a)	Text + DISC	42.96 ± 17.73	126.56 ± 37.93	0.1777 ± 0.2206
	IndexTTS Deng et al. (2025)	Voice + DISC	71.19 ± 13.77	123.65 ± 43.99	0.2548 ± 0.2560
	Vevo2 Zhang et al. (2026)	Voice + DISC	75.11 ± 8.20	136.00 ± 45.09	0.1242 ± 0.1806
	CookVoice				
	$S_{\mathcal{T}} + \mathcal{P}_{disc}$	Text + DISC	60.78 ± 14.86$_{(+41.48\%)}$	119.56 ± 50.73$_{(\downarrow 5.53\%)}$	0.3929 ± 0.2395$_{(+121.10\%)}$
	$S_{\mathcal{T}} + \mathcal{P}_{cont}$	Text + CONT	73.09 ± 16.38	88.85 ± 38.56	0.6416 ± 0.2011
	$S_{\mathcal{V}} + \mathcal{P}_{disc}$	Voice + DISC	87.47 ± 3.77$_{(+16.46\%)}$	99.05 ± 43.77$_{(\downarrow 19.89\%)}$	0.5330 ± 0.2338$_{(+109.18\%)}$
	$S_{\mathcal{V}} + \mathcal{P}_{cont}$	Voice + CONT	91.65 ± 2.98	74.98 ± 24.53	0.7102 ± 0.1714
TTSV	DiffSinger Liu et al. (2022)	Text + DISC	75.81 ± 5.81	100.48 ± 26.23	0.4580 ± 0.1857
	StyleSinger Zhang et al. (2024a)	Voice + DISC	82.00 ± 5.40	101.82 ± 25.79	0.4856 ± 0.1518
	TCSinger Zhang et al. (2024b)	Voice + DISC	87.53 ± 5.02	92.50 ± 22.01	0.5643 ± 0.1749
	Vevo1.5 Zhang et al. (2025)	Voice + CONT	86.55 ± 7.18	79.56 ± 28.69	0.7125 ± 0.1615
	Vevo2 Zhang et al. (2026)	Voice + CONT	88.09 ± 3.56	90.09 ± 26.94	0.6242 ± 0.1479
	CookVoice				
	$S_{\mathcal{T}} + \mathcal{P}_{disc}$	Text + DISC	85.75 ± 5.41$_{(+13.11\%)}$	97.73 ± 27.37$_{(\downarrow 2.74\%)}$	0.5383 ± 0.1825$_{(+17.53\%)}$
	$S_{\mathcal{T}} + \mathcal{P}_{cont}$	Text + CONT	72.51 ± 18.02	68.12 ± 20.69	0.8401 ± 0.0655
	$S_{\mathcal{V}} + \mathcal{P}_{disc}$	Voice + DISC	93.72 ± 2.00$_{(+7.07\%)}$	95.01 ± 28.43$_{(\uparrow 2.71\%)}$	0.5704 ± 0.1685$_{(+1.08\%)}$
	$S_{\mathcal{V}} + \mathcal{P}_{cont}$	Voice + CONT	95.00 ± 1.72$_{(+7.84\%)}$	56.96 ± 14.76$_{(\downarrow 28.41\%)}$	0.8425 ± 0.0711$_{(+18.25\%)}$

Table 6: Quality evaluation in terms of Mean Opinion Score (**MOS**), Melody Comparative Mean Opinion Score (**MC-MOS**) and Word Error Rate (**WER**) for Chinese (**CH**) and English (**EN**).

Task	Model	Style + Prosody	MOS ↑	MC-MOS ↑	WER-CH ↓	WER-EN ↓
TTS	CosyVoice Du et al. (2024)	Text + DISC	4.30 ± 0.90	–	5.97 ± 7.53	3.14 ± 6.11
	CosyVoice Du et al. (2024)	Voice + DISC	4.05 ± 1.07	–	11.27 ± 12.40	5.79 ± 9.85
	F5-TTS Chen et al. (2025)	Voice + DISC	4.35 ± 0.96	–	5.23 ± 7.45	2.34 ± 4.67
	ParaStyleTTS Lou et al. (2025a)	Text + DISC	4.03 ± 1.01	–	9.67 ± 8.25	5.21 ± 6.95
	IndexTTS Deng et al. (2025)	Voice + DISC	4.42 ± 0.80	–	4.16 ± 6.63	2.88 ± 6.04
	Vevo2 Zhang et al. (2026)	Voice + DISC	4.30 ± 0.75	–	6.35 ± 6.73	3.82 ± 6.47
	CookVoice					
	$S_{\mathcal{T}} + \mathcal{P}_{disc}$	Text + DISC	3.45 ± 1.14	–	9.48 ± 11.88	6.07 ± 12.35
	$S_{\mathcal{T}} + \mathcal{P}_{cont}$	Text + CONT	3.80 ± 1.08	–	9.03 ± 10.98	5.23 ± 8.76
	$S_{\mathcal{V}} + \mathcal{P}_{disc}$	Voice + DISC	3.60 ± 1.02	–	8.66 ± 10.14	4.41 ± 7.40
	$S_{\mathcal{V}} + \mathcal{P}_{cont}$	Voice + CONT	3.98 ± 0.91	–	8.13 ± 10.07	4.19 ± 7.45
	Ground Truth		4.05 ± 0.92	–	6.33 ± 9.33	3.39 ± 7.07
TTSV	DiffSinger Liu et al. (2022)	Text + DISC	2.80 ± 0.81	-0.80 ± 1.66	17.79 ± 14.16	–
	StyleSinger Zhang et al. (2024a)	Voice + DISC	3.20 ± 1.08	-0.40 ± 1.61	11.31 ± 7.68	–
	TCSinger Zhang et al. (2024b)	Voice + DISC	2.40 ± 1.07	-0.95 ± 1.41	9.88 ± 7.23	29.03 ± 19.31
	Vevo1.5 Zhang et al. (2025)	Voice + CONT	3.33 ± 1.29	-0.74 ± 1.66	8.85 ± 11.09	23.62 ± 14.67
	Vevo2 Zhang et al. (2026)	Voice + CONT	3.42 ± 1.30	0.11 ± 1.76	4.66 ± 6.04	11.37 ± 9.80
	CookVoice					
	$S_{\mathcal{T}} + \mathcal{P}_{disc}$	Text + DISC	2.40 ± 0.89	-1.39 ± 1.24	11.71 ± 10.44	19.56 ± 14.47
	$S_{\mathcal{T}} + \mathcal{P}_{cont}$	Text + CONT	3.20 ± 0.93	-0.19 ± 1.44	9.79 ± 9.26	21.10 ± 17.81
	$S_{\mathcal{V}} + \mathcal{P}_{disc}$	Voice + DISC	2.80 ± 1.00	-0.94 ± 1.41	11.45 ± 9.14	22.53 ± 16.92
	$S_{\mathcal{V}} + \mathcal{P}_{cont}$	Voice + CONT	3.40 ± 0.97	0.28 ± 1.44	10.01 ± 9.70	24.37 ± 17.23
	Ground Truth		3.95 ± 0.89	–	5.01 ± 4.76	14.22 ± 12.93

Table 7: Intelligibility and prosodic evaluation comparison on Chinese and English subsets. Word Error Rate (**WER**), Phoneme Error Rate (**PhoER**), Prosody Error Rate (**ProER**) ↓ indicates lower is better.

		Chinese			English		
Task	**Model**	**WER**	**PhoER**	**ProER**	**WER**	**PhoER**	**ProER**
TTS	CosyVoice Du et al. (2024)	8.62 ± 10.60	3.48 ± 12.19	2.63 ± 7.80	4.46 ± 8.30	3.93 ± 8.35	2.65 ± 6.71
	F5-TTS Chen et al. (2025)	5.23 ± 7.45	0.47 ± 1.67	0.63 ± 1.78	2.34 ± 4.67	1.63 ± 3.32	0.98 ± 1.96
	ParaStyleTTS Lou et al. (2025a)	9.67 ± 8.25	4.63 ± 6.70	3.27 ± 6.54	5.21 ± 6.95	5.40 ± 7.42	1.82 ± 2.80
	IndexTTS Deng et al. (2025)	4.16 ± 6.63	0.35 ± 1.60	0.35 ± 1.60	2.88 ± 6.04	2.04 ± 5.26	1.29 ± 4.07
	Vevo2 Zhang et al. (2026)	6.35 ± 6.73	1.56 ± 2.93	1.17 ± 2.62	3.82 ± 6.47	2.72 ± 6.01	1.62 ± 4.33
	CookVoice						
	$S_{\mathcal{T}} + \mathcal{P}_{disc}$	9.48 ± 11.88	4.22 ± 10.75	3.79 ± 9.17	6.07 ± 12.35	5.80 ± 12.43	3.05 ± 6.09
	$S_{\mathcal{T}} + \mathcal{P}_{cont}$	9.03 ± 10.98	3.07 ± 8.65	2.69 ± 6.74	5.23 ± 8.76	4.44 ± 8.85	2.64 ± 5.20
	$S_{\mathcal{V}} + \mathcal{P}_{disc}$	8.66 ± 10.14	4.28 ± 9.66	3.13 ± 6.97	4.41 ± 7.40	3.97 ± 7.21	1.90 ± 3.50
	$S_{\mathcal{V}} + \mathcal{P}_{cont}$	8.13 ± 10.07	3.78 ± 9.19	3.17 ± 6.69	4.19 ± 7.45	3.77 ± 7.81	1.47 ± 2.97
	Ground Truth	6.33 ± 9.33	2.01 ± 5.46	1.78 ± 4.99	3.39 ± 7.07	2.84 ± 7.14	1.31 ± 3.58
TTSV	DiffSinger Liu et al. (2022)	17.79 ± 14.16	16.43 ± 13.99	17.04 ± 14.21	–	–	–
	StyleSinger Zhang et al. (2024a)	11.31 ± 7.68	10.43 ± 8.87	10.05 ± 7.54	–	–	–
	TCSinger Zhang et al. (2024b)	9.88 ± 7.23	7.61 ± 6.15	7.40 ± 6.26	29.03 ± 19.31	31.45 ± 21.85	16.11 ± 17.33
	Vevo1.5 Zhang et al. (2025)	8.85 ± 11.09	8.40 ± 10.92	9.34 ± 9.43	23.62 ± 14.67	24.91 ± 15.17	11.23 ± 8.04
	Vevo2 Zhang et al. (2026)	4.66 ± 6.04	2.89 ± 4.96	4.21 ± 5.00	11.37 ± 9.80	10.84 ± 9.61	4.74 ± 4.48
	CookVoice						
	$S_{\mathcal{T}} + \mathcal{P}_{disc}$	11.71 ± 10.44	11.08 ± 11.79	10.80 ± 8.96	19.56 ± 14.47	20.81 ± 15.71	8.79 ± 6.72
	$S_{\mathcal{T}} + \mathcal{P}_{cont}$	9.79 ± 9.26	9.82 ± 12.25	8.92 ± 7.27	21.10 ± 17.81	22.17 ± 19.71	10.05 ± 8.91
	$S_{\mathcal{V}} + \mathcal{P}_{disc}$	11.45 ± 9.14	11.15 ± 10.12	10.01 ± 7.93	22.53 ± 16.92	23.32 ± 19.20	10.53 ± 7.88
	$S_{\mathcal{V}} + \mathcal{P}_{cont}$	10.01 ± 9.70	8.65 ± 11.99	8.78 ± 7.83	24.37 ± 17.23	24.68 ± 20.09	10.54 ± 6.70
	Ground Truth	5.01 ± 4.76	3.06 ± 5.33	4.99 ± 5.98	14.22 ± 12.93	13.16 ± 10.92	7.26 ± 8.00