

TTS²: Test-Time Scaling for Flow-Based Text-to-Speech via Multi-stage Reward Modeling on Posterior Mean Projections

Tung Luong Nguyen^{1,2*}, Duy Le-Tuan Nguyen^{1*}, Nhat Hong Tran^{1,2*},
Nhat Minh Le¹, Van Hai Do^{1,3†}

¹Viettel AI, Viettel Group

²Hanoi University of Science and Technology

³Thuyloi University

haidv21@viettel.com.vn

Abstract

While training-time scaling has revolutionized generative models, inference-time compute scaling remains largely unexplored in speech synthesis. In this work, we investigate methods to effectively utilize computation during inference to improve the performance of non-autoregressive Text-to-Speech (TTS) systems, specifically those based on Conditional Flow Matching (CFM). By analyzing CFM synthesis behavior, we identify distinct performance plateaus during the denoising process. To overcome these limitations, we propose TTS² - a multi-stage, test-time compute framework that optimizes TTS quality. By combining Ordinary Differential Equation (ODE) and Stochastic Differential Equation (SDE) generation dynamics with stochastic search and specialized verifiers, our approach enables guided trajectory optimization for timbre consistency, content accuracy, and audio fidelity. To the best of our knowledge, this work is the first to formalize and study test-time scaling for non-autoregressive speech synthesis. Source code, models, and audio demos are publicly available at ([Source code](#)).

1 Introduction

Concurrent with the significant advancements in generative models across various domains, recent speech synthesis technologies have produced high-quality models trained on large-scale datasets, capable of delivering stable, controllable, and exceptionally human-like speech (Chen et al., 2025; Eskimez et al., 2024; Zhou et al., 2025; Du et al., 2025). A primary driver of this breakthrough is the principle of scaling laws, which correlate performance gains with increases in training data, compute, and model parameters (Kaplan et al., 2020). However, as the diminishing availability of high-quality data and the exponential growth of required computational

resources cause traditional scaling laws to reach saturation, the focus is shifting away from purely scaling training efforts (Snell et al., 2025). Specifically, recent research demonstrates that allocating more compute during inference through sophisticated sampling and search techniques can significantly enhance response accuracy and reasoning quality, offering a new trajectory for test-time performance gains.

Inspired by the success of Large Language Models (LLMs), many state-of-the-art Text-to-Speech (TTS) systems utilize autoregressive modeling over discrete speech units, achieving impressive zero-shot capabilities (Xie et al., 2025; Wang et al., 2025). However, the inherent limitations of autoregressive approaches, including high inference latency and exposure bias, remain significant challenges. Furthermore, the inherent limitations of existing speech tokenizers raise concerns regarding the representational quality of discrete speech tokens. In contrast, non-autoregressive (NAR) TTS models, which model directly in continuous space, offer the advantage of fast, parallel inference while effectively balancing quality and speed (Ju et al., 2024). Progressing from diffusion models to Conditional Flow Matching (CFM) (Lipman et al., 2023) with Optimal Transport paths, these NAR models have achieved high-fidelity speech synthesis with remarkable precision (Chen et al., 2025; Eskimez et al., 2024).

Under the next-token prediction paradigm, test-time computation in LLMs is relatively straightforward, as each discrete step produces a complete, interpretable token. In contrast, diffusion-based models operate through a denoising mechanism where intermediate outputs are inherently noisy, making it difficult to evaluate the model’s progress mid-inference. Furthermore, previous research has shown that simply increasing the number of denoising steps is an ineffective way to scale inference compute, as performance typically

*Equal contribution

†Corresponding author

plateaus after a certain number of function evaluations (NFEs) (Karras et al., 2022). Motivated by the significant limitations in current research regarding inference-time scaling for TTS, this work investigates methods to effectively utilize compute through various exploration and exploitation techniques, specifically within Conditional Flow Matching based models. To the best of our knowledge, we are the first to explore test-time scaling methods in NAR speech synthesis. We first provide an analysis of speech synthesis behavior in CFM models throughout the denoising process. Building on these insights, we propose a multi-stage test-time compute framework that optimizes both performance and efficiency. This is achieved by leveraging the flexibility of search algorithms and differential equation solvers, integrated with a diverse array of verifiers specifically addressing timbre consistency, content accuracy, and overall audio fidelity, to evaluate and refine output quality throughout the scaling process.

Our main contributions are summarized as follows:

- We formalize the problem of inference-time scaling in flow-based TTS models and analyze critical design aspects for performance optimization.
- We propose a multi-stage test-time scaling framework that flexibly integrates various search strategies and verifiers to optimize multiple dimensions of audio quality.
- Experiments across diverse evaluation sets demonstrate the superiority of our approach in achieving timbre consistency, content precision, and high-fidelity speech output.

2 Background

2.1 Conditional Flow Matching for TTS

Recent zero-shot text-to-speech models increasingly adopt Conditional Flow Matching (Chen et al., 2025; Eskimez et al., 2024; Le et al., 2023), which learns a continuous vector field that deterministically transports a simple prior to the empirical data distribution (Lipman et al., 2023). Formally, let x_1 denote the target mel-spectrogram and $x_0 \sim \mathcal{N}(0, I)$ be a sample drawn from a standard Gaussian prior. To ensure computational efficiency, these models typically utilize a linear interpolant, where the intermediate state x_t at time $t \in [0, 1]$ is defined as:

$$x_t = (1 - t)x_0 + tx_1 \quad (1)$$

The generative process is governed by an Ordinary Differential Equation (ODE), where a neural network $v_\theta(x_t, t)$ is trained via regression to approximate the target vector field:

$$dx_t = v_\theta(x_t, t)dt \quad (2)$$

During inference, solving this ODE maps the initial noise x_0 to the fully generated speech representation x_1 . Crucially, unlike the fully-formed discrete tokens in autoregressive models, any intermediate state x_t is a noisy, incomplete mel-spectrogram. These sub-terminal states lack resolved acoustic or linguistic properties, complicating the application of standard speech evaluation metrics during the generation trajectory.

2.2 Test-Time Scaling and Search Strategy

Test-time scaling (Muennighoff et al., 2025) significantly enhances output quality in discrete domains, such as autoregressive speech language models (Wu et al., 2026; Ye et al., 2025), where algorithms like Monte Carlo Tree Search can branch at fully-formed tokens under the guidance of step-wise reward models. In contrast, for diffusion or flow-based models, the straightforward approach of scaling the number of denoising steps—has shown diminishing returns. Therefore, alternative search-based approaches must be considered.

Applying analogous search mechanisms to continuous, CFM-based TTS poses a fundamental architectural challenge. As established, a standard ODE solver yields a single, deterministic trajectory from a specific noise prior x_0 to the target mel-spectrogram x_1 , inherently lacking the branching points required to construct a search space. Although a fixed mapping exists between the initial noise and the final sample, the primary source of explicit randomness lies in the sampling of these initial priors. Consequently, scaling the number of initial noise candidates becomes the only viable axis for test-time scaling within a purely ODE-based framework.

To enable test-time exploration, recent literature (Kim et al., 2025a) reformulates the ODE into a Stochastic Differential Equation (SDE). By injecting stochasticity—typically via a standard Wiener process dw_t —the SDE framework allows a single initial noise state to diverge into a diverse distribution of generative paths.

2.3 Verifiers as Guidance for Inference-Time Search

In the context of test-time scaling, verifiers serve as feedback mechanisms that guide the search toward outputs better aligned with preferred characteristics. Specifically, given a verifier $f(\mathbf{x}_1) : \mathbb{R}^D \rightarrow \mathbb{R}^+$ which evaluates the quality of a candidate and returns a scalar score, our objective is to identify a sampling trajectory that maximizes the value of $f(\mathbf{x}_1)$ (Ma et al., 2025).

In CFM-based TTS models, output mel-spectrograms are converted into waveforms via a vocoder. For particle sampling, which requires evaluating noisy steps for selection, the clean data manifold is estimated using the posterior mean derived from Tweedie’s formula (Robbins, 1992). By utilizing the learned velocity field $v_\theta(x_t, t)$, we project the noisy intermediate representation x_t forward to its estimated clean output, denoted as $\hat{x}_1(t)$:

$$\hat{x}_1(t) = x_t + (1 - t)v_\theta(x_t, t) \quad (3)$$

3 Methodology

To address the limitations of single-strategy test-time search, we propose **TTS²**, a method for **Test-Time Scaling for Flow-matching based Text-to-Speech**. First, we introduce an inference-time SDE conversion to enable particle sampling in flow models. By tracking the evolution of intermediate noisy representations, we identify distinct phases of feature convergence. As illustrated in Fig. 1, TTS² leverages this by reformulating the generation flow into a branching search space, transitioning through three targeted phases to sequentially optimize semantics, timbre, and acoustic fidelity.

3.1 Stochastic Continuous Flow via Linear Interpolant

The standard inference procedure for CFM relies on an ODE solver to integrate the learned velocity field $v_\theta(x_t, t)$. However, as discussed in Sec.2.2, the deterministic nature of this sampling process precludes the branching mechanisms necessary for test-time particle search.

To construct a robust search space while maintaining simplicity and compatibility with off-the-shelf CFM models, we reformulate the deterministic ODE into an SDE built directly upon the standard linear interpolant. Following recent adaptations of flow models for stochastic generation (Kim et al., 2025a), we inject stochasticity into the linear flow. This allows a single initial state to diverge

into a diverse manifold of candidate trajectories. The modified sampling process is defined by the SDE:

$$dx_t = \left[v_\theta(x_t, t) - \frac{1}{2}g(t)^2 \nabla_{x_t} \log p_t(x_t) \right] dt + g(t)dw_t \quad (4)$$

where $g(t)$ is a time-dependent diffusion coefficient controlling the injected noise, $\nabla_{x_t} \log p_t(x_t)$ is the score function (which can be approximated via the velocity field), and w_t represents a standard Wiener process. By retaining the linear interpolant underlying the base ODE, this SDE provides the critical exploratory noise dw_t necessary to drive our multi-stage search. This stochastic formulation serves as the generative engine across all phases that use SDE-based sampling of our work.

3.2 The dynamics of projected intermediate representations

To guide the generative process, we analyze speech signal dynamics to characterize model behavior throughout the denoising process. Since the x_t are too noisy for direct evaluation, we project them onto the clean data manifold via the posterior mean (Sec.2.3, Eq.3) and transform them into waveforms using the Vocos vocoder. As shown in Fig. 2 for the F5-TTS (Chen et al., 2025) model, evaluating these states reveals distinct convergence phases. Linguistic intelligibility (WER) converges first during the initial flow steps. Next, speaker identity (SIM) establishes and plateaus during the intermediate stages. Finally, acoustic details are refined during the later stages of the integration process.

3.3 Multi-Stage Reward Trajectory

Based on empirical observations above of generation flow, speech features converge sequentially. Specifically, linguistic intelligibility (WER) establishes first at t_1 , followed by speaker identity (SIM-o) at t_2 , defining the phase boundaries $0 < t_1 < t_2 < 1$. To dynamically align our computational strategy with this progression, we initialize N parallel trajectories, denoted as $\{x_t^i\}_{i=1}^N$, and divide the generative flow $t \in [0, 1]$ into three targeted operational phases, as summarized in Algorithm 1.

Phase 1: SDE Warm-up ($0 \leq t < t_1$) In the initial stage, the primary objective is establishing the general semantic and phonetic structure. Because the Gaussian prior dominates early representations, estimated clean signals are excessively noisy and

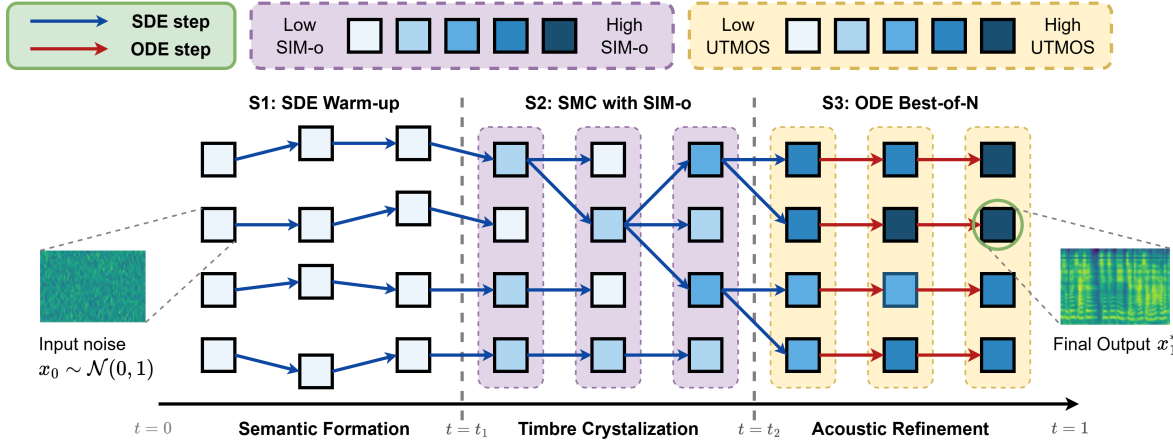


Figure 1: **Overview of TTS²**. Compute is aligned across the flow via a branching SDE search space. Trajectories begin with an unguided warm-up, refine speaker features through an SMC search, and converge using an ODE Best-of-N selection to finalize acoustic details.

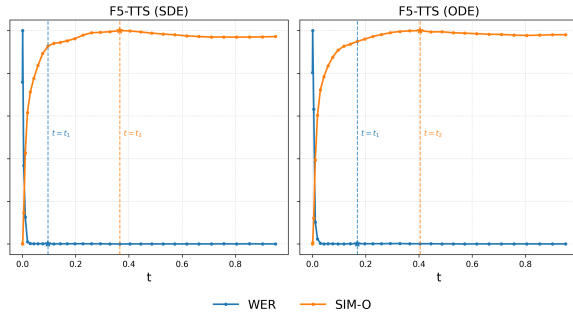


Figure 2: Estimated speech metrics for the F5-TTS (Chen et al., 2025) model evaluated on projected intermediate representations $\hat{x}_1(t)$ across the generative flow using the SDE solver, illustrating the sequential convergence of semantic and speaker identity features.

reward gradients unstable. Thus, N particles are propagated purely via the unguided SDE to branch out and increase candidate diversity. This exploration allows the model to generate the underlying semantic information before converging at t_1 .

Phase 2: Timbre-guided Sequential Monte Carlo ($t_1 \leq t < t_2$) Beyond t_1 , semantic features stabilize, and the model shifts to optimizing speaker characteristics. Since SIM converges next at t_2 , we deploy Sequential Monte Carlo (SMC) steering targeted at speaker identity. At each step, we estimate each particle’s clean data manifold via the Tweedie projection for our linear interpolant (similar to Eq. 3). Decoding, evaluating, and resampling these projected states based on SIM scores prunes suboptimal paths, thereby concentrating SDE resources on trajectories that best capture the target speaker’s timbre.

Algorithm 1 TTS²: Test-Time Scaling for Flow-Matching Text-to-Speech

Require: Speaker ref. s_{ref} , Num of Particles N , boundaries t_1, t_2 , Step size Δt
Ensure: Final high-fidelity waveform y^*
1: **Init:** $X_0 = \{x_0^1, \dots, x_0^N\} \sim \mathcal{N}(0, \mathbf{I})$
2: **for** $t = 0$ **to** $1 - \Delta t$ **step** Δt **do**
3: **if** $0 \leq t < t_1$ **then** // Phase 1: SDE Warm-up
4: **for** $i = 1 \dots N$ **do**
5: $x_{t+\Delta t}^i \leftarrow \text{SDE_step}(x_t^i)$ // algorithm 3
6: **else if** $t_1 \leq t < t_2$ **then** // Phase 2: Timbre SMC
7: **for** $i = 1 \dots N$ **do**
8: $x_{t+\Delta t}^i \leftarrow \text{SDE_step}(x_t^i)$
9: $\hat{x}_1^i \leftarrow \text{Tweedie}(x_{t+\Delta t}^i, c)$
10: $\hat{y}^i \leftarrow \text{Vocoder}(\hat{x}_1^i)$
11: $w^i \leftarrow \exp(\text{SIM}(\hat{y}^i, s_{\text{ref}})/\beta)$
12: Normalize $W = \{w^1, \dots, w^N\}$ // section D.2
13: Resample $X_{t+\Delta t}$ using W
14: **else** // Phase 3: ODE Refinement
15: **for** $i = 1 \dots N$ **do**
16: $x_{t+\Delta t}^i \leftarrow \text{ODE_step}(x_t^i)$
17: **Terminal Selection:** // Best-of-N
18: **for** $i = 1 \dots N$ **do**
19: $y^i \leftarrow \text{Vocoder}(x_1^i)$
20: **return** $\arg \max_{y^i} \text{UTMOS}(y^i)$

Phase 3: Acoustic Refinement via Deterministic ODE ($t_2 \leq t \leq 1$) By the time the flow reaches t_2 , semantic and speaker information are established, making further SDE branching unnecessary. Injecting stochasticity at this late stage merely introduces noise and computational overhead, risking performance degradation. Therefore, we transition to a deterministic ODE solver to integrate the remaining flow and converge fine-grained acoustic details. Finally, at $t = 1$, we perform a terminal Best-of-N (BoN) selection using UTMOS to isolate the highest-fidelity waveform, y^* .

4 Experimental Setup

4.1 Models and Datasets

To demonstrate the architectural generalization of the TTS² framework, we evaluate our multi-stage search across representative CFM-TTS models: F5-TTS, E2-TTS (Eskimez et al., 2024), and Matcha-TTS (Mehta et al., 2024). All models utilize their officially released pre-trained checkpoints without additional fine-tuning. For zero-shot evaluation of F5-TTS and E2-TTS, we utilize the Seed-TTS test-en (Anastassiou et al., 2024) and Seed-TTS test-zh to assess performance on English and Mandarin, alongside the standard LibriSpeech-PC (Meister et al., 2023) test-clean benchmark. For Matcha-TTS, we conduct evaluations on the LJSpeech (Ito and Johnson, 2017) dataset.

4.2 Evaluation Metrics

We follow established zero-shot TTS protocols (Chen et al., 2025) for our objective evaluation of our method:

- **Linguistic Intelligibility:** Word Error Rate (WER) evaluated utilizing the Whisper-large-v3 model (Radford et al., 2023).
- **Speaker Similarity:** Cosine similarity computed between the speaker embeddings of the reference prompt and the synthesized utterance (SIM-o), extracted via WavLM-large (Chen et al., 2022).
- **Acoustic Naturalness:** Objectively quantified using the UTMOS (Saeki et al., 2022) predictor to estimate Mean Opinion Score.

For subjective evaluation, comparative Mean Opinion Score (CMOS) and Similarity Mean Opinion Score (SMOS) are assessed by human listeners on the LibriSpeech-PC test-clean subset. Detailed crowd-sourcing and evaluation procedures are deferred to Appendix C. The computational budget is evaluated using the Real-Time Factor (RTF), a standard metric for assessing real-world TTS applications. The RTF measures execution efficiency by calculating the ratio of the model’s audio generation time to the actual duration of the synthesized speech.

4.3 Baselines and Search Configurations

We employ two common approaches as baselines for test-time scaling: Best-of-N using an ODE solver and Sequential Monte Carlo using an SDE

solver. BoN selects the single candidate with the highest reward from N independently generated samples. Meanwhile, SMC (Kim et al., 2025b) maintains a population of N particles and performs resampling based on importance weights at each denoising step. To maintain a constant inference budget of 256 NFEs across all experiments, we set $N = 8$ and the number of denoising steps to 32. We determine the phase boundaries t_1 and t_2 for each model based on the convergence analysis presented in Section 3.2; the specific values and our search-based validation procedure are detailed in Appendix A. Additionally, to verify robustness across different synthesis backends, we expand our evaluation beyond the default Vocos (Siuzdak, 2024) vocoder to include alternative vocoders such as BigVGAN (gil Lee et al., 2023) and HiFi-GAN (Kong et al., 2020).

5 Experimental Results

Table 1, 2, 3 present the detailed evaluation results across three key dimensions: objective metrics, subjective assessments, and computational performance speed. The following subsections detail our observations drawn from these results.

5.1 Objective Evaluation

We first observe that given a fixed computational budget—measured by the Number of Function Evaluations for flow-based models such as E2-TTS and F5-TTS, solely increasing the number of integration steps yields only marginal improvements. For instance, scaling from 32 to 256 steps for E2-TTS on LibriSpeech-PC improves the WER by 0.14%. In contrast, allocating the same computational budget to search-based methods, such as BoN sampling and SMC, produces significantly greater gains. However, optimizing generation with respect to a single verifier introduces distinct trade-offs between evaluation metrics, reflecting both the inherent biases of individual verifiers and the risk of reward hacking.

Specifically, approaches utilizing a single SIM-o verifier achieve the highest speaker similarity scores—with BoN SIM-o on E2-TTS reaching a SIM-o score of 0.76 on LibriSpeech-PC—but typically at the expense of a degraded WER. Conversely, UTMOS-optimized variants improve speech naturalness, with F5-TTS using BoN UTMOS reaching a UTMOS of 4.08, while sacrificing speaker similarity. Additionally, given the same

Table 1: Evaluation results on LibriSpeech-PC test-clean, Seed-TTS test-en and test-zh. Best in **bold**, second-best underlined.

Model	Steps	N	LibriSpeech-PC test-clean			Seed-TTS test-en			Seed-TTS test-zh		
			WER (%) ↓	SIM-o ↑	UTMOS ↑	WER (%) ↓	SIM-o ↑	UTMOS ↑	WER (%) ↓	SIM-o ↑	UTMOS ↑
Ground Truth	—	—	2.23	0.69	4.09	2.06	0.73	3.53	1.26	0.76	2.78
E2-TTS	256	—	2.49	0.71	3.75	1.77	0.73	3.63	1.80	0.78	2.94
E2-TTS	32	—	2.63	0.72	3.72	1.79	0.73	3.60	<u>1.82</u>	0.78	2.91
+ BoN SIM-o	32	8	2.63	0.76	3.71	1.93	0.76	3.44	<u>1.86</u>	0.80	2.81
+ BoN UTMOS	32	8	2.29	0.72	4.00	1.85	0.72	<u>3.74</u>	1.84	0.76	<u>3.03</u>
+ SMC SIM-o	32	8	2.37	<u>0.73</u>	3.76	2.16	0.74	3.66	2.09	0.81	2.96
+ SMC UTMOS	32	8	<u>2.26</u>	0.72	3.83	2.08	0.73	3.71	1.96	0.77	3.02
+ TTS ² (ours)	32	8	2.08	<u>0.73</u>	4.01	<u>1.78</u>	<u>0.75</u>	3.91	1.89	<u>0.80</u>	3.07
F5-TTS	256	—	2.41	0.68	3.92	1.68	0.68	3.78	<u>1.54</u>	0.75	2.93
F5-TTS	32	—	2.36	0.66	3.91	1.70	0.67	3.76	1.59	0.75	2.96
+ BoN SIM-o	32	8	2.24	0.71	3.88	1.78	0.72	3.68	1.63	0.78	2.92
+ BoN UTMOS	32	8	2.22	0.67	4.08	1.65	0.66	3.89	1.64	0.75	3.12
+ SMC SIM-o	32	8	2.39	0.71	3.90	1.72	<u>0.70</u>	3.71	1.67	0.77	3.00
+ SMC UTMOS	32	8	2.28	0.66	4.02	1.64	0.66	3.80	1.70	0.74	3.06
+ TTS ² (ours)	32	8	2.18	<u>0.68</u>	4.10	1.63	<u>0.70</u>	3.95	1.42	0.78	3.15

verifier and computational budget, BoN consistently outperforms SMC. This suggests that when employing a single test-time search strategy, preserving the natural Ordinary Differential Equation solver trajectory of the CFM model yields superior generation quality.

Our multi-stage framework, TTS², which combines search with step-wise guidance by applying different verifiers at specific stages of the integration process, consistently outperforms both BoN and SMC under the same inference budget. It achieves the best overall balance across WER and UTMOS while maintaining competitive SIM-o performance. This success is achieved thanks to multi-reward balancing, an equilibrium that no single-verifier baseline manages to attain simultaneously. Notably, TTS² with E2-TTS achieves a WER of 2.08% on LibriSpeech-PC, falling below even the ground-truth WER of 2.23% without being directly optimized for intelligibility. Regarding multilingual capabilities, TTS² with F5-TTS achieves superior performance across all three metrics on both the Seed-TTS *test-en* and *test-zh* benchmarks. Specifically, its WER of 1.42% on *test-zh* represents a substantial reduction over the baseline F5-TTS score of 1.59%.

5.2 Subjective Evaluation

Subjective evaluations further corroborate these findings. As shown in Table 2, TTS² achieves a CMOS of 1.43 and an SMOS of 3.56, outperforming all BoN and SMC variants by a substantial margin and exceeding the next-best method by over 0.85 CMOS points. This confirms that the improvements in perceptual quality are meaningful and consistent with the objective evaluation metrics.

5.3 Computational Efficiency

We further analyze the computational efficiency of each method in Table 3. BoN variants achieve the lowest RTFs, at 1.64 and 1.66 respectively, because candidate generation is straightforward and highly parallelizable. In contrast, SMC incurs a substantially higher RTF of up to 7.00 due to its sequential resampling steps and frequent queries to the reward model. TTS² achieves a moderate RTF of 2.84; while this introduces a slight overhead compared to the 256-step baseline model, it remains significantly more efficient than SMC. Considering that TTS² consistently outperforms the alternative approaches across most objective and subjective metrics, this trade-off between efficiency and quality is highly favorable. This positions TTS² as a practical and effective framework for high-quality, inference-time scaling.

Overall, the evaluation results demonstrate that while test-time scaling inherently improves output quality, the strategic allocation of computation is just as critical as the total budget itself. Single-verifier baselines reveal a fundamental tension: optimizing for a single aspect of speech quality tends to degrade others. In contrast, TTS² resolves this tension through staged computation that progressively refines distinct quality dimensions. By shifting from a uniform increase in NFEs to a multi-stage framework, we demonstrate that performance is maximized when search strategies and verifiers are tailored to specific phases of the generative trajectory. This finding suggests a broader principle for generative models with structured sampling paths: the structured composition of test-time compute across distinct stages can be more impactful than its overall magnitude.

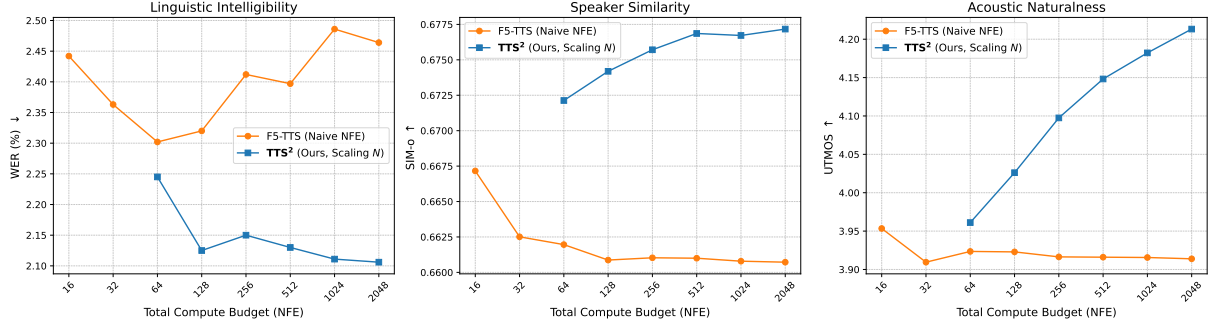


Figure 3: **Compute scaling behavior.** Naive ODE-step scaling quickly saturates, while particle scaling in TTS² provides sustained gains in speaker similarity and acoustic naturalness.

Table 2: Subjective evaluation results (CMOS and SMOS) on the LibriSpeech-PC *test-clean* subset. Comparative MOS evaluates preference against the respective 32-step base model. 95% confidence intervals are provided.

Model	Steps	N	CMOS ↑	SMOS ↑
F5-TTS	32	–	0.31 ± 0.14	2.86 ± 0.17
+ BoN SIM-o	32	8	0.41 ± 0.14	3.29 ± 0.16
+ BoN UTMOS	32	8	0.53 ± 0.16	2.97 ± 0.17
+ SMC SIM-o	32	8	0.48 ± 0.13	3.02 ± 0.19
+ SMC UTMOS	32	8	0.51 ± 0.13	2.93 ± 0.19
+ TTS ² (ours)	32	8	1.43 ± 0.15	3.59 ± 0.18

Table 3: Real-time factor (RTF) comparison under different inference-time scaling strategies. RTF is computed for 10s speech on a single NVIDIA A100. Lower is better.

Model	Steps	N	RTF ↓
F5-TTS	256	–	1.73
+ BoN SIM-o	32	8	1.66
+ BoN UTMOS	32	8	1.64
+ SMC SIM-o	32	8	7.00
+ SMC UTMOS	32	8	6.04
+ TTS ² (ours)	32	8	2.84

5.4 Impact of Inference-Time Scaling

To isolate the efficacy of our multi-stage compute allocation, we compare TTS² against a naive denoising scaling baseline. We scale the inference budget of the F5-TTS base model along two distinct axes: uniformly increasing the Ordinary Differential Equation (ODE) integration steps, and increasing the particle population size (N) within our framework, using a fixed denoising steps of 32.

As illustrated in Figure 3, naive ODE scaling (from 2 to 2048 steps) plateaus rapidly. While acoustic naturalness (UTMOS) and intelligibility (WER) improve up to 32 steps, injecting additional computation into a deterministic trajectory yields marginal gains and occasionally degrades speaker similarity (SIM-o) due to over-integration artifacts.

Conversely, given the same NFEs, scaling the particle population $N \in \{2, 4, 8, \dots, 64\}$ in TTS² unlocks sustained improvements across all metrics. By expanding the search space during the SDE Warm-up and Timbre Crystallization phases, the Sequential Monte Carlo (SMC) verifiers successfully isolate intermediate representations with superior SIM-o and UTMOS scores without compromising WER. Notably, all tested particle budgets yield strict Pareto improvements over a compute-equivalent 256-step naive ODE baseline, validating dynamic branching as a fundamentally superior scaling dimension. Detailed numerical results for scaling parameters are provided in Appendix B.

5.5 Vocoder Robustness

To ensure our framework optimizes for general speech properties rather than overfitting to a single vocoder’s biases, we conducted an ablation study by replacing Vocos with BigVGAN, a structurally different GAN-based vocoder. The experiment is conducted on F5-TTS, keeping the same TTS² pipeline and NFE budget. The results in Table 4 show that TTS² consistently improves over the baseline regardless of the vocoder choice, suggesting the reward signal reflects genuine mel-spectrogram quality.

Table 4: Objective evaluation metrics for F5-TTS with different vocoders.

Method	Vocoder	WER (%) ↓	SIM-o ↑	UTMOS ↑
Baseline (ODE, NFE=256)	Vocos	2.41	0.68	3.92
Baseline (ODE, NFE=256)	BigVGAN	2.04	0.65	3.64
TTS ² (NFE=256)	Vocos	2.18	0.68	4.10
TTS ² (NFE=256)	BigVGAN	2.04	0.67	3.75

5.6 Cross-Architecture Generalization

To demonstrate that the TTS² framework generalizes beyond the E2-TTS and F5-TTS architec-

tures, we applied it to Matcha-TTS, an optimal-transport CFM model. We tested on the LJSpeech dataset with both Vocos and HiFiGAN vocoders at 40 NFEs. As shown in Table 5, TTS² improves over the Matcha-TTS baseline across almost all metrics, for both vocoders. This further confirms that the benefits of our approach are architecture-agnostic and robust to vocoder variations. Detailed experimental setup and evaluation procedures for this single-speaker configuration are provided in Appendix G.

Table 5: Evaluation results on Matcha-TTS using Vocos and HiFiGAN.

Method	Vocoder	WER (%) ↓	SIM-o ↑	UTMOS ↑
Matcha-TTS (ODE)	Vocos	3.01	0.82	4.10
Matcha-TTS + TTS ²	Vocos	3.01	0.83	4.25
Matcha-TTS (ODE)	HiFiGAN	2.01	0.84	4.14
Matcha-TTS + TTS ²	HiFiGAN	2.01	0.85	4.25

6 Conclusion

We study inference-time scaling for non-autoregressive text-to-speech models based on Conditional Flow Matching. Our analysis shows that increasing the number of denoising steps alone provides limited gains due to performance plateaus during the synthesis trajectory. To address this limitation, we introduce a multi-stage test-time compute framework that combines search-based inference with stage-wise verifier guidance. This approach enables more effective allocation of inference computation and improves timbre consistency, content accuracy, and speech quality under the same compute budget. Experimental results demonstrate consistent improvements over standard inference and single-verifier search methods. These findings highlight inference-time scaling as a promising direction for advancing NAR speech synthesis without additional training.

Limitations

Although TTS² demonstrates promising results, several limitations warrant consideration for future improvements:

Static Phase Boundaries Our current framework relies on static, dataset-level phase boundaries (t_1 and t_2) determined via average convergence plateaus. While effective for the models and datasets evaluated, these static boundaries may not be strictly optimal for every individual utterance, particularly for highly out-of-domain prompts or

zero-shot application to new languages. Developing an adaptive, per-utterance method for estimating these transitions dynamically during inference remains an important direction for future research.

Reward Overfitting and Verifier Trade-offs

While both our objective metrics and human evaluations confirm consistent quality improvements, reliance on specific verifiers invariably introduces the risk of reward overfitting—where the search algorithm might exploit the verifier’s blind spots rather than genuinely improving acoustic quality. Furthermore, as demonstrated by our single-verifier baselines, there is an inherent trade-off when optimizing strictly for naturalness (UTMOS) versus speaker similarity (SIM-o). Although TTS² successfully balances these dimensions through its multi-stage design, the framework’s efficacy is fundamentally tied to the quality and complementarity of the chosen verifiers.

Integration of Speech Language Models While recent test-time scaling approaches in text and visual generation leverage Large Language Models (LLMs) or Vision-Language Models (VLMs) as verifiers, our framework currently relies on task-specific discriminative models. Exploring the integration of unified, large-scale Speech Language Models (SLMs) as holistic verifiers could potentially mitigate reward overfitting and provide a more comprehensive signal for trajectory optimization.

Acknowledgments

This work was supported by Viettel AI. The authors gratefully acknowledge their financial backing and resources. Furthermore, we wish to thank the human evaluators whose participation in our perceptual studies was instrumental to the subjective validation of this method.

References

- Philip Anastassiou, Jiawei Chen, Jitong Chen, Yanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, and 1 others. 2024. Seed-tts: A family of high-quality versatile speech generation models. *arXiv preprint arXiv:2406.02430*.
- Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, and 1 others. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE*

- Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.
- Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. 2025. 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)*, pages 6255–6271.
- Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Chongjia Ni, Xian Shi, and 1 others. 2025. Cosyvoice 3: Towards in-the-wild speech generation via scaling-up and post-training. *arXiv preprint arXiv:2505.17589*.
- Sefik Emre Eskimez, Xiaofei Wang, Manthan Thakker, Canrun Li, Chung-Hsien Tsai, Zhen Xiao, Hemin Yang, Zirun Zhu, Min Tang, Xu Tan, and 1 others. 2024. E2 tts: Embarrassingly easy fully non-autoregressive zero-shot tts. In *2024 IEEE spoken language technology workshop (SLT)*, pages 682–689. IEEE.
- Sang gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. 2023. **BigVGAN: A universal neural vocoder with large-scale training**. In *The Eleventh International Conference on Learning Representations*.
- Keith Ito and Linda Johnson. 2017. The lj speech dataset.
- Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, and 1 others. 2024. Naturspeech 3: zero-shot speech synthesis with factorized codec and diffusion models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 22605–22623.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.
- Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. 2022. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577.
- Jaihoon Kim, TaeHoon Yoon, Jisung Hwang, and Minhyuk Sung. 2025a. Inference-time scaling for flow models via stochastic generation and rollover budget forcing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Sunwoo Kim, Minkyu Kim, and Dongmin Park. 2025b. Test-time alignment of diffusion models without reward over-optimization. In *The Thirteenth International Conference on Learning Representations*.
- Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033.
- Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and 1 others. 2023. Voicebox: Text-guided multilingual universal speech generation at scale. *Advances in neural information processing systems*, 36:14005–14034.
- Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2023. Flow matching for generative modeling. In *11th International Conference on Learning Representations, ICLR 2023*.
- Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and 1 others. 2025. Scaling inference time compute for diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2523–2534. IEEE.
- Shivam Mehta, Ruibo Tu, Jonas Beskow, Éva Székely, and Gustav Eje Henter. 2024. Matcha-TTS: A fast TTS architecture with conditional flow matching. In *Proc. ICASSP*.
- Aleksandr Meister, Matvei Novikov, Nikolay Karpov, Evelina Bakhturina, Vitaly Lavrukhin, and Boris Ginsburg. 2023. Librispeech-pc: Benchmark for evaluation of punctuation and capitalization capabilities of end-to-end asr models. In *2023 IEEE automatic speech recognition and understanding workshop (ASRU)*, pages 1–7. IEEE.
- Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. **s1: Simple test-time scaling**. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20275–20321, Suzhou, China. Association for Computational Linguistics.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.
- Herbert E Robbins. 1992. An empirical bayes approach to statistics. In *Breakthroughs in Statistics: Foundations and basic theory*, pages 388–394. Springer.
- Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. Utmos: Utokyo-sarulab system for voicemos challenge 2022. *Interspeech 2022*.
- Hubert Siuzdak. 2024. Vocos: Closing the gap between time-domain and fourier-based neural vocoders for

high-quality audio synthesis. In *International Conference on Learning Representations*, volume 2024, pages 25719–25733.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. [Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning](#). In *The Thirteenth International Conference on Learning Representations*.

Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, and 1 others. 2025. Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens. *arXiv preprint arXiv:2503.01710*.

Yuan-Kuei Wu, Yang Liu, Yiteng Huang, Zhaojun Yang, Haibin Wu, Ruizhe Huang, Yi-TeEthan Hsu, Shuyu Kong, Ming Sun, Florian Metze, and Li Wan. 2026. [Slm-tta: A framework for test-time adaptation of generative spoken language models](#). In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 17777–17781.

Kun Xie, Feiyu Shen, Junjie Li, Fenglong Xie, Xu Tang, and Yao Hu. 2025. Firedtts-2: Towards long conversational speech generation for podcast and chatbot. *arXiv preprint arXiv:2509.02020*.

Zhen Ye, Xinfa Zhu, Chi-Min Chan, Xinsheng Wang, Xu Tan, Jiahe Lei, Yi Peng, Haohe Liu, Yizhu Jin, Zheqi Dai, and 1 others. 2025. Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis. *arXiv preprint arXiv:2502.04128*.

Yixuan Zhou, Guoyang Zeng, Xin Liu, Xiang Li, Renjie Yu, Ziyang Wang, Runchuan Ye, Weiyue Sun, Jiancheng Gui, Kehan Li, and 1 others. 2025. Voxcpm: Tokenizer-free tts for context-aware speech generation and true-to-life voice cloning. *arXiv preprint arXiv:2509.24650*.

A Empirical Justification of t_1 and t_2

The phase boundaries t_1 and t_2 are critical hyperparameters that dictate the transition between semantic formation, timbre crystallization, and acoustic refinement within our multi-stage framework.

Heuristic Initialization. As analyzed in Section 3.2 and illustrated in Figure 2, the convergence of linguistic and speaker features across the generative flow follows a predictable trajectory. Therefore, our primary method for selecting t_1 and t_2 relies on a static heuristic derived directly from these convergence plateaus. Specifically, for F5-TTS, we set $t_1 = 0.3$ and $t_2 = 0.6$, while for E2-TTS, we use $t_1 = 0.1$ and $t_2 = 0.4$.

Search-Based Validation. To rigorously validate these heuristic boundaries, we investigated an

alternating, metric-driven search procedure adapted from the IntMeanFlow framework. Initializing with our heuristic peaks, the search alternates iteratively (Algorithm 2): we first freeze t_1 to optimize t_2 , and subsequently freeze t_2 to refine t_1 , utilizing SIM-o as the objective reward function over a held-out validation subset of the Emilia dataset.

Algorithm 2 Alternating Search for Phase Boundaries

Require: Base model, validation set $\mathcal{V}$, heuristic initializations $t_1^{(0)}, t_2^{(0)}$, total inference steps $K = 32$

Ensure: Optimized boundaries t_1^*, t_2^*

- 1: Let $\mathcal{T} = \{\frac{1}{K}, \frac{2}{K}, \dots, \frac{K-1}{K}\}$ be the set of valid discrete timesteps
 - 2: Initialize $t_1 \leftarrow t_1^{(0)}, t_2 \leftarrow t_2^{(0)}$
 - 3: **while** not converged **do**
// Step 1: Freeze t_1 , search t_2 over subsequent timesteps
 4: $t_2 \leftarrow \arg \max_{t \in \mathcal{T}, t > t_1} \text{SIM}(\text{TTS}^2(t_1, t; \mathcal{V}))$
// Step 2: Freeze t_2 , search t_1 over preceding timesteps
 5: $t_1 \leftarrow \arg \max_{t \in \mathcal{T}, t < t_2} \text{SIM}(\text{TTS}^2(t, t_2; \mathcal{V}))$
 6: **return** t_1, t_2
-

Our empirical evaluations revealed that this dynamic search converged to boundaries nearly identical to our static heuristic initialization. Furthermore, the computationally intensive search yielded no statistically significant improvement in the final acoustic fidelity or speaker similarity over the heuristic baseline. Consequently, we employ the static, peak-based initialization in all primary experiments, as it achieves optimal generation quality while avoiding the prohibitive computational overhead of a dedicated test-time hyperparameter search.

B Detailed Scaling Results

Table 6 provides the comprehensive numerical results for the inference-time scaling ablation discussed in Section 5.4. We compare the impact of scaling the total compute budget via two distinct methods: uniformly increasing the number of function evaluations (NFE) using a deterministic ODE solver, and increasing the particle population size (N) within the TTS² framework. For TTS², the base integration steps are kept constant at 32.

C Subjective Evaluation Details

To evaluate the perceptual quality and speaker similarity of our proposed method, we follow the subjective evaluation protocols established in F5-TTS (Chen et al., 2025), specifically measuring Comparative Mean Opinion Score (CMOS) and Similarity Mean Opinion Score (SMOS).

Table 6: Objective evaluation metrics for F5-TTS comparing deterministic ODE step scaling (NFE) against particle population scaling (N) in TTS². Best values for each compute tier are highlighted.

NFE	N	WER (%) ↓	SIM-o ↑	UTMOS ↑
Naive ODE Scaling				
4	-	20.58	0.47	1.89
8	-	2.84	0.63	3.52
16	-	2.44	0.67	3.95
32	-	2.36	0.66	3.91
64	-	2.30	0.66	3.92
128	-	2.32	0.66	3.92
256	-	2.41	0.66	3.92
512	-	2.40	0.66	3.92
1024	-	2.49	0.66	3.92
2048	-	2.46	0.66	3.91
TTS² Population Scaling (Ours)				
32	2	2.25	0.67	3.96
32	4	2.13	0.67	4.03
32	8	2.15	0.67	4.10
32	16	2.14	0.68	4.15
32	32	2.11	0.68	4.18
32	64	2.11	0.68	4.21

For the evaluation, we sample 30 synthesized utterances per model configuration from the LibriSpeech-PC test-clean subset. We recruited 20 fluent English speakers to participate in the double-blind listening tests, with instructions to use headphones in a quiet environment. To assess acoustic naturalness and prosody, we measure CMOS by asking listeners to rate their preference between paired audio samples (our proposed model versus the baseline) on a standard 7-point scale ranging from -3 to $+3$. To assess zero-shot speaker modeling accuracy, we measure SMOS by presenting listeners with the ground-truth reference prompt followed by the synthesized utterance, asking them to rate the speaker similarity on a 5-point scale from 1 to 5.

D Detailed Algorithm Description

D.1 Stochastic Denoising Step

Phase 1 in algorithm 1 applies one stochastic denoising update to each particle, as summarized in algorithm 3. Starting from the current particle $\mathbf{x}_t^i$, we first evaluate the model velocity, convert it to the corresponding score under the chosen interpolant, form the reverse-time drift, and then apply one Euler–Maruyama update. The diffusion strength follows the square schedule used in our implementation, so stronger noise is injected at earlier steps

Algorithm 3 SDE_STEP: one-step stochastic denoising

Require: particle $\mathbf{x}_t^i$, step t , step size Δt , flow model u_θ

Ensure: updated particle $\mathbf{x}_{t+\Delta t}^i$

- 1: $\mathbf{v}_t^i \leftarrow u_\theta(\mathbf{x}_t^i, t)$
- 2: $\mathbf{s}_t^i \leftarrow \frac{-(1-t)\mathbf{v}_t^i - \mathbf{x}_t^i}{t}$
- 3: $g_t \leftarrow g(t)$
- 4: $\mathbf{f}_t^i \leftarrow \mathbf{v}_t^i - \frac{g_t^2}{2}\mathbf{s}_t^i$
- 5: Sample $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
- 6: $\mathbf{x}_{t+\Delta t}^i \leftarrow \mathbf{x}_t^i + \mathbf{f}_t^i \Delta t + g_t \sqrt{\Delta t} \mathbf{z}$

and gradually vanishes near the end of sampling.

D.2 SMC Weight Normalization and Resampling

During the Timbre SMC phase ($t_1 < t \leq t_2$), the particle weights are updated using the reward increment scaled by the temperature parameter β , and then normalized as

$$\tilde{w}^i = \frac{w^i}{\sum_{j=1}^N w^j}. \quad (5)$$

Following standard Sequential Monte Carlo practice (Kim et al., 2025b), resampling is triggered when the effective sample size (ESS) falls below a threshold τ :

$$\text{ESS} = \frac{\left(\sum_{j=1}^N \tilde{w}^j\right)^2}{\sum_{i=1}^N (\tilde{w}^i)^2} < \tau. \quad (6)$$

When the ESS criterion is satisfied, particles with larger normalized weights are more likely to be selected as ancestors, while low-weight particles are discarded. After resampling, all particle weights are reset uniformly.

D.3 Implementation Details

For all TTS² experiments, we use classifier-free guidance strength $c = 2.0$, sway sampling coefficient -1.0 , generation speed 1.0 , and target RMS 0.1 .

For the procedure in algorithm 1, we fix the weight-temperature parameter to $\beta = 1.0$, and the ESS resampling threshold in eq. (6) to $\tau = 0.5$.

In the stochastic phase, we use the square diffusion schedule in algorithm 3, namely $g(t) = \lambda(1-t)^2$. Following prior work (Kim et al., 2025a), we fix the diffusion scale to $\lambda = 0.03$.

E Latency Breakdown

Table 7 provides a detailed latency breakdown of the TTS² framework across its three phases. As ex-

pected, the SMC search phase dominates the computational budget due to the evaluation of multiple particles.

Table 7: Per-phase latency breakdown for TTS² ($N = 8$).

Phase	NFE	N	RTF	% of Total
Warm-up (SDE)	13	8	1.63	23.4%
SMC Search	7	8	6.97	53.8%
ODE Refinement	12	8	1.76	22.9%
Total	32	8	2.84	100%

F Statistical Significance

To rigorously validate the objective improvements of TTS², we conducted Wilcoxon Signed-Rank tests comparing our approach against all baseline models on both UTMOS and SIM-o metrics. The results of these statistical analyses are presented in Table 8 and Table 9.

As shown in Table 8, TTS² significantly outperforms all baseline methods in terms of overall speech naturalness (UTMOS), yielding p -values < 0.001 across all pairwise comparisons. The substantial win/loss ratios against both the base F5-TTS model (892 wins to 235 losses) and single-verifier variants further demonstrate a robust and consistent margin of improvement. Moreover, the 95% confidence interval of our approach (4.098 ± 0.016) shows no overlap with the majority of the baselines, establishing a clear statistical superiority in generating high-fidelity audio.

Table 8: UTMOS Statistical Analysis

Method (B)	Overall Mean & 95% CI	Wilcoxon p-value	Win / Loss (Ours vs. B)	Significance
Ours (TTS²)	4.098 $\pm$ 0.016	—	—	—
F5-TTS (Base)	3.916 $\pm$ 0.021	$p < 0.001$	892 / 235	Significant
bon_utm	4.080 $\pm$ 0.016	$p < 0.001$	891 / 236	Significant
smc_utm	4.021 $\pm$ 0.023	$p < 0.001$	607 / 520	Significant
smc_sim	3.904 $\pm$ 0.022	$p < 0.001$	909 / 218	Significant
bon_sim	3.878 $\pm$ 0.022	$p < 0.001$	964 / 163	Significant

Regarding speaker similarity (Table 9), TTS² achieves significant improvements over the base model, BoN UTMOS, and SMC UTMOS baselines. While methods exclusively optimized for SIM-o (such as BoN SIM-o and SMC SIM-o) understandably achieve higher speaker similarity scores, they suffer from severe degradation in speech naturalness, as evidenced by their poor UTMOS performance. In contrast, TTS² maintains highly competitive SIM-o performance while strictly outperforming all variants in naturalness, successfully resolving the inherent trade-off between speaker

similarity and acoustic fidelity. These objective statistical findings corroborate the subjective CMOS and SMOS evaluations presented in Appendix C, firmly validating the effectiveness of our proposed multi-stage verification framework.

Table 9: SIM Statistical Analysis

Method (B)	Overall Mean & 95% CI	Wilcoxon p-value	Win / Loss (Ours vs. B)	Significance
Ours (TTS²)	0.682 $\pm$ 0.005	—	—	—
F5-TTS (Base)	0.678 $\pm$ 0.005	$p = 0.0011$	627 / 500	Significant
smc_utm	0.663 $\pm$ 0.004	$p < 0.001$	802 / 325	Significant
bon_utm	0.665 $\pm$ 0.005	$p < 0.001$	987 / 140	Significant
bon_sim	0.713 $\pm$ 0.004	$p < 0.001$	276 / 851	Significant (B Wins)
smc_sim	0.712 $\pm$ 0.005	$p < 0.001$	240 / 887	Significant (B Wins)

G Detailed Setup for Cross-Architecture Generalization

Since Matcha-TTS and the LJSpeech dataset are designed for single-speaker synthesis rather than zero-shot voice cloning, our experimental setup and evaluation protocols differ from those used for F5-TTS and E2-TTS. Specifically, we utilize the official LJSpeech test split to synthesize utterances based on their corresponding ground-truth transcripts. To measure speaker similarity (SIM-o) in this context, we compute the cosine similarity between the embeddings of the synthesized utterance and the ground-truth audio of the identical test utterance, which serves as the reference. This setup allows us to rigorously evaluate the model’s ability to reconstruct the target speaker’s voice in a standard single-speaker scenario.