SinDiffPhase: High-Quality Phase Estimation with Ultra-Fast Single-Step Diffusion

Yifei Ni*†‡, Andong Li†‡, Lingling Dai†‡, Erwei Yin§¶, Qunping Ni‖ and Chengshi Zheng†‡
* Harbin Engineering University, Harbin, China
† Institute of Acoustics, Chinese Academy of Sciences, Beijing, China
‡ Chinese Academy of Sciences, Beijing, China
§ Defense Innovation Institute, Academy of Military Sciences (AMS), Beijing, China
¶ Tianjin Artificial Intelligence Innovation Center (TAIIC), Tianjin, China
‖ TianJin 712 Communication and Broadcasting Co., Ltd, Tianjin, China

***Abstract*—Phase estimation is a classical problem in speech processing. Recently, diffusion-based phase estimation methods have achieved promising performance due to the powerful capability of generative models. Nonetheless, they usually suffer from inherent drawbacks like excessive computational complexity and relatively slow inference speed. To this end, this paper proposes SinDiffPhase, a novel single-step diffusion-based phase estimation model. Through unifying the setting of discrete-time steps to 1, and substituting the original score-matching loss function with direct optimization of the phase and its differential forms along the time and frequency dimensions, the proposed model achieves phase generation in a single step for the first time. Extensive experiments on the VCTK corpus show that while achieving around 35.1× faster inference speed than DiffPhase-small, a state-of-the-art diffusion-based phase estimation model, the proposed model achieves competitive performance and notably outperforms existing traditional and DNN-based baselines in various objective metrics, fully demonstrating the potential of the proposed method in real-time applications.**

I. Introduction

Speech processing technology is essential in our daily life, with applications in communication, intelligent assistants, hearing aids, and so on. High-quality speech output is essential for enhancing user experience. Most speech processing tasks are performed in the time-frequency domain and often involve using short-time Fourier transforms (STFT) to convert signals into complex-valued spectra, where it is widely accepted that the amplitude and phase components jointly affect speech processing quality [1], [2]. Phase estimation, also known as phase retrieval or phase reconstruction, attempts to recover the speech phase spectrum only with the information of the amplitude spectrum [3]. It is widely applied to speech tasks such as speech enhancement and speech synthesis, especially when phase information is missing [4], [5].

Traditional phase estimation methods and their variants [6], [7], [8] generally use alternating projections but suffer from slow convergence and rely on random phase initialization. Phase estimation methods in the optical domain have been adapted to speech processing, showing some advantages over general acoustic estimation methods [9].Recently, deep learning has introduced new paradigms for phase estimation. For example, iterative optimization frameworks integrate deep neural networks (DNNs) into the iterative process to improve phase estimation accuracy [10], [11]. In addition, by incorporating adversarial training, generative adversarial networks (GANs) can effectively model the phase distribution [12]. Despite the promising performance, existing discriminative approaches can suffer from performance and generalization limitations due to their direct mapping characteristic [13].

In more recent years, diffusion generative models have demonstrated significant advantages in speech generation tasks by simulating the forward-reverse diffusion process of spectra, capturing complex time-frequency dependencies [14], [15], [16], [17]. In [18], Peer et al. proposed DiffPhase and its lightweight variant DiffPhase-small, where the diffusion score model [17] is adopted to implicitly reconstruct the phase via real and imaginary (RI) spectra estimation. This method achieves state-of-the-art (SOTA) performance in perceptual evaluation of speech quality (PESQ) [19] and extended short-time objective intelligibility (ESTOI) [20]. However, similar to traditional iterative methods, its multistep diffusion mechanism ensures precise phase estimation at the cost of higher model complexity and reduced generation speed. Moreover, while reducing the number of iterations in a naive way can boost speed, it leads to a severe degradation in estimation quality. Accordingly, the core challenge lies in balancing the phase estimation quality and the processing speed of the model.

To remedy this, this paper proposes SinDiffPhase, a novel single-step diffusion model based on DiffPhase-small, for the phase estimation task. In particular, the single-step strategy is adopted for fast phase estimation. Besides, we replace the original score matching loss with the phase distance constraint, as well as its time and frequency differential formats. In this way, we enforce the model to efficiently capture the complicated patterns of the phase spectrum. To the best of our knowledge, this is the first time that diffusion-based phase estimation has been achieved within a single step. The proposed model notably surpasses various traditional and DNN

Chengshi Zheng is the corresponding author (corr.). E-mail: cszheng@mail.ioa.ac.cn

979-8-3315-7206-8/25/$31.00

methods in both estimation accuracy and processing efficiency. Ablation studies further validate the importance of the two key strategies proposed in this paper.

II. Related Work

A. Problem Formulation

The input signal $x(t)$ can be transformed into a complex spectrogram $X(k,l)$ via STFT:

$$X(k,l) = \sum_{n=0}^{N-1} x(lR+n)w(n)e^{-j\frac{2\pi kn}{K}}, \tag{1}$$

where k denotes the frequency bin, l is the frame index, $w(n)$ is the window function, N is the window length, R is the hop size, and K is the FFT size (typically $K \geq N$). The complex spectrogram can be decomposed into magnitude and phase spectra as:

$$X = Ae^{j\phi}, \tag{2}$$

where A and ϕ represent the magnitude and phase spectra, respectively.

This study focuses on estimating the phase spectrum ϕ from a given magnitude spectrum A under noise-free conditions, formulating the problem as an optimization task:

$$\min_X \|X - P_c(X)\|_{Fro}^2 \text{ s.t. } X \in \mathcal{A}, \tag{3}$$

$$P_c(X) = \text{STFT}(\text{iSTFT}(X)), \tag{4}$$

where $\|\cdot\|_{\text{Fro}}$ is the Frobenius norm, P_c is metric projections onto the consistent spectrograms C, $\mathcal{A}$ is the set of spectrograms with magnitudes matching A, and iSTFT denotes the inverse STFT.

B. Griffin-Lim Algorithm (GLA)

GLA [6] iteratively refines phase estimates to satisfy magnitude constraints:

$$X^{[m+1]} = P_{\mathcal{C}}\big(P_{\mathcal{A}}\big(X^{[m]}\big)\big), \tag{5}$$

$$P_{\mathcal{A}}(X) = A \odot X \oslash |X|, \tag{6}$$

where $X^{[m]}$ is the updated complex spectrogram at the iteration index of m, $P_{\mathcal{A}}$ is a metric projection onto the magnitude-consistent sets $\mathcal{A}$, and $\odot$ and $\oslash$ denote element-wise multiplication and division, respectively.

GLA is computationally simple but prone to local optima. It also requires a large number of iterative steps, leading to slow convergence.

C. Fast Griffin-Lim Algorithm (FGLA)

Introducing an adaptive step size to GLA, FGLA [7] was proposed:

$$Y^{[m+1]} = P_c(P_{\mathcal{A}}(X^{[m]})), \tag{7}$$

$$X^{[m+1]} = Y^{[m+1]} + \alpha_m(Y^{[m+1]} - Y^{[m]}), \tag{8}$$

where α_m is an acceleration parameter. Studies show that FGLA achieves higher phase estimation quality than GLA, with $\alpha_m = 0.99$ yielding optimal results.

Despite improvements, similar to GLA, it essentially relies on multiple iterations of complex spectra and may converge to local optima.

D. Deep Griffin-Lim Iteration (DeGLI)

DeGLI [10] combines DNNs with GLA, approximating GLA iterations using weight-shared sub-blocks. The number of iterations can be adjusted by changing the depth of the model. However, it does not break away from the essence of iterative computation. Each iteration even takes longer than that of GLA, and the quality of output still requires improvement.

E. Neural Speech Phase Prediction (NSPP)

NSPP [21] predicts the wrapped phase spectra directly through a neural network with parallel real and imaginary output layers. The network is then trained by minimizing the phase loss function. This model features superior training and generation speeds, and outperforms other methods in signal-to-noise ratio (SNR) and root mean squared error of F0 (F0-RMSE). Nevertheless, its phase estimation quality still allows for further improvement.

F. Generative Diffusion-Based STFT Phase Retrieval (DiffPhase)

Using Ornstein-Uhlenbeck variance exploding (OUVE) stochastic differential equations (SDEs) [22], DiffPhase [18] models the phase information loss and recovery as the forward-reverse process of noise addition and removal.

1) Forward Process:

$$dx_t = f(x_t, y, t)dt + g(t)dw, \tag{9}$$

$$f(x_t, y, t) = \gamma(y - x_t), \tag{10}$$

$$g(t) = \sigma_{\text{min}}\left(\frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}\right)^t \sqrt{2\ln\left(\frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}\right)}. \tag{11}$$

Here, x_t denotes the current spectrogram, y represents the spectrogram with the phase initialized as all-zeros. $f(x_t, y, t)$ is the drift coefficient, $g(t)$ is the diffusion coefficient, w is the Wiener process, and $t \in [0, T]$ indicates the time step of the current diffusion process, with the forward diffusion corresponding to time steps from 0 to T. γ, σ_{min} and σ_{max} are all scalar constant parameters.

2) Reverse Process:

$$dx_t = \left[-f(x_t, y, t) + (g(t))^2 s_\theta(x_t, t)\right] dt + g(t)d\bar{w}. \tag{12}$$

Here, $s_\theta(x_t, t)$ represents the score function, and $\overline{w}$ denotes the time-reversed Wiener process. The reverse process corresponds to time steps t from T to 0.

By discretizing t into N discrete-time steps (with $N = 1000$ for training and $N = 30$ for generation), this model is trained via denoising score matching, which employs a score loss function to optimize $s_\theta(x_t, t)$.

DiffPhase and its lightweight variant, DiffPhase-small (which has around 60% reduction in parameters compared to DiffPhase), achieve SOTA performance in both PESQ and ESTOI, but inevitably sacrifice speed for quality with numerous diffusion steps.

III. PROPOSED METHOD

Existing phase estimation methods face a quality-speed trade-off, where DiffPhase and DiffPhase-small achieve high quality via inefficient multistep diffusion. To address it, this study proposes SinDiffPhase, where the single-step inference is applied based on DiffPhase-small. Specifically, we constrain the single step in both forward and reverse processes to mitigate the latent exposure bias [23], which is common in diffusion models. Moreover, by incorporating the phase optimization target, we enable more accurate estimation and efficient processing.

A. Model Structure

1) Discrete-Time Training and Generation: To reduce the training and generation time of the diffusion model for real-time processing, we unify the discrete-time steps N in both training and testing stages to 1 for single-step phase estimation. The forward and reverse processes are illustrated in Fig. 1.

a) Forward Process: The clean complex spectrum x_0 is directly corrupted to generate a zero-phase complex spectrum x_T by adding noise:

$$x_T = \mu(x_0, y, T) + \sigma(T)z, \tag{13}$$

$$\mu(x_0, y, T) = e^{-\gamma T}x_0 + (1 - e^{-\gamma T})y, \tag{14}$$

$$\sigma(T)^2 = \frac{\sigma_{\min}^2 \left(\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^{2T} - e^{-2\gamma T} \right) \ln\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)}{\gamma + \ln\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)}, \tag{15}$$

where $z \sim \mathcal{N}_c(0; \mathbf{I})$ is a sample of standard Gaussian noise. $\mu(x_0, y, T)$ and $\sigma(T)$ are the mean and variance of the complex spectrum, respectively. y is the spectrum after phase zero-initialization, same as x_T.

b) Reverse Process: According to (10), drift coefficient $f(x_T, y, T)$ used in SinDiffPhase is equal to 0. Using the Euler-Maruyama method, we recover x_0 from x_T as

$$x_0' = x_T + g(T)^2 s_\theta(x_T, T)T, \tag{16}$$

$$g(T) = \sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^T \sqrt{2\ln\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)}, \tag{17}$$

where x_0' is the estimated clean complex spectrum, $g(T)$ is the diffusion coefficient, and $s_\theta(x_T, T)$ is the score function estimated by the network. By reformulating the forward and reverse processes, the proposed method enables the efficient estimation of the target phase spectrum within one step.

2) Loss Function: Given that the original score loss function of DiffPhase-small relies on iterative refinement, directly retaining it may fail to capture global phase correlation. To effectively exploit the complicated patterns of the phase spectrum, motivated by [21], we opt to directly optimize phase with the anti-wrapping phase loss, including both instantaneous phase and the differential counterparts along the time and frequency axes. Besides, we empirically observe that the loss value during training is usually quite small, which can impede the training and performance. Consequently, a clip-level loss strategy is adopted instead of the T-F bin-level version, which calculates the mean loss for each individual sample. Specifically, the estimated phase spectrum is first averaged within each speech clip, and then summed along the batch dimension. This simple design enables the model to capture the global correlation of phases directly. Similar to [21], the anti-wrapping function f_{AW} is defined as:

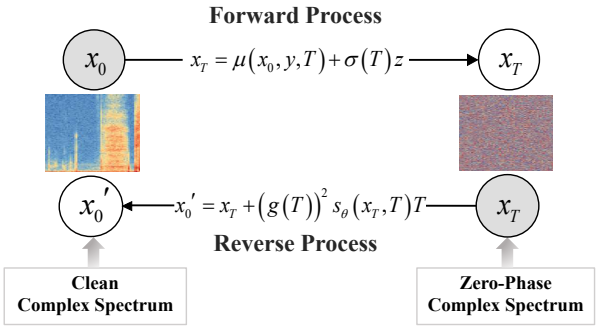

Fig. 1. Forward and reverse diffusion processes of SinDiffPhase (single-step mechanism highlighted)

$$f_{AW}(x) = \left| x - 2\pi \cdot round\left(\frac{x}{2\pi}\right) \right|, \tag{18}$$

where $round(\cdot)$ represents the rounding, x is the unwrapped phase, and $|\cdot|$ denotes the absolute error. f_{AW} converts x to the wrapped domain. The phase loss function used in SinDiffPhase is formulated as:

$$\mathcal{L} = \mathcal{L}_{\text{IP}} + \mathcal{L}_{\text{GD}} + \mathcal{L}_{\text{IAF}}, \tag{19}$$

$$\mathcal{L}_{\text{IP}} = \mathbb{E}_{(\hat{\mathbf{P}}, \mathbf{P})} \sum_{n=1}^{b} \overline{f_{AW}\left(\hat{P}_n - P_n\right)}, \tag{20}$$

$$\mathcal{L}_{\text{GD}} = \mathbb{E}_{(\Delta_{\text{DF}}\hat{\mathbf{P}}, \Delta_{\text{DF}}\mathbf{P})} \sum_{n=1}^{b} \overline{f_{AW}\left(\Delta_{\text{DF}}\hat{P}_n - \Delta_{\text{DF}}P_n\right)}, \tag{21}$$

$$\mathcal{L}_{\text{IAF}} = \mathbb{E}_{(\Delta_{\text{DT}}\hat{\mathbf{P}}, \Delta_{\text{DT}}\mathbf{P})} \sum_{n=1}^{b} \overline{f_{AW}\left(\Delta_{\text{DT}}\hat{P}_n - \Delta_{\text{DT}}P_n\right)}, \tag{22}$$

where $\hat{P}_n$ and P_n denote the estimated and ground-truth wrapped phase spectra of the n-th speech clip in the batch, b is the batch size, $\mathbb{E}_{(a,b)}$ denotes the expectation of a and b, and $\bar{Y}$ denotes the element-wise mean of matrix Y. Δ_{DF} and Δ_{DT} represent the differential operations along the frequency and time axes, respectively. The instantaneous phase loss $\mathcal{L}_{\text{IP}}$ measures the discrepancy between the estimated and ground-truth wrapped phases, while the group delay loss $\mathcal{L}_{\text{GD}}$ and the instantaneous angular frequency loss $\mathcal{L}_{\text{IAF}}$ optimize the frequency and time continuity of the estimated wrapped phase spectrum, respectively.

B. Standardized Protocol for Training and Testing

Similar to [18], the NCSN++ network is adopted as the score estimator. During training, the noisy spectrum x_T and the condition spectrum y are fed into the DNN to output the score function. Reverse sampling via (16) yields the estimated clean complex spectrum x'_0, whose phase component is then used to compute the loss function against the ground-truth phase. During generation, the magnitude spectrum is inputted to the network, which outputs the estimated score function, followed by inverse sampling, and obtaining the reconstructed phase from the estimated RI spectrum.

IV. EXPERIMENTS

A. Experimental Data and Configuration

Experimental data is sourced from the VCTK corpus [24], which originally comprises 109 speakers (47 males and 61 females). All speech samples from the corpus are downsampled to 16 kHz. Specifically, speech data from 101 speakers are split into a training set (consisting of 9398 clips with uniform length) and a validation set (including 47 clips with uniform length). Furthermore, the test set is constructed using 2937 utterances of varying lengths, which are extracted from the remaining 8 speakers. For spectrum extraction, a 32 ms window size, an 8 ms window shift, and 512 FFT points are utilized. Evaluation metrics for PESQ, ESTOI, and scale-invariant signal-to-distortion ratio (SI-SDR) [25] are employed. Higher values of these three metrics indicate better performance in phase estimation quality, reflecting abilities to improve speech quality, intelligibility, and reduce signal distortion, respectively. The real-time factor (RTF) serves as the speed metric, where lower RTF values correspond to faster generation, and RTF < 1 enables real-time application. Experiments are conducted on an NVIDIA GeForce RTX 4090 GPU, except for RTF testing, which runs on an Intel Xeon(R) Gold 6430 CPU.

B. Cross-Method Comparative Experiments

Objective experiments are conducted to compare the performance of the proposed SinDiffPhase and other phase estimation methods. The descriptions of the methods for comparison are as follows:

- SinDiffPhase: The proposed model requires only one iteration for training and generation. A random 256-frame speech slice is used, yielding a 256×256 input spectrogram. Hyperparameters are set as $\sigma_{\text{min}} = 0.05$, $\sigma_{\text{max}} = 0.5$, $\gamma = 1.5$, and $T = 1$. The batch size is 8, with 32640 samples per speech clip, and training spans 1700 epochs.
- GLA: Each speech file is processed with varying iteration counts n.
- FGLA: FGLA uses an acceleration parameter $\alpha_m = 0.99$ and iterates n times.
- DeGLI: Spectral iterations of DeGLI are controlled via identical n stacked training modules, with a batch size of 8. Training is conducted for 200 epochs.

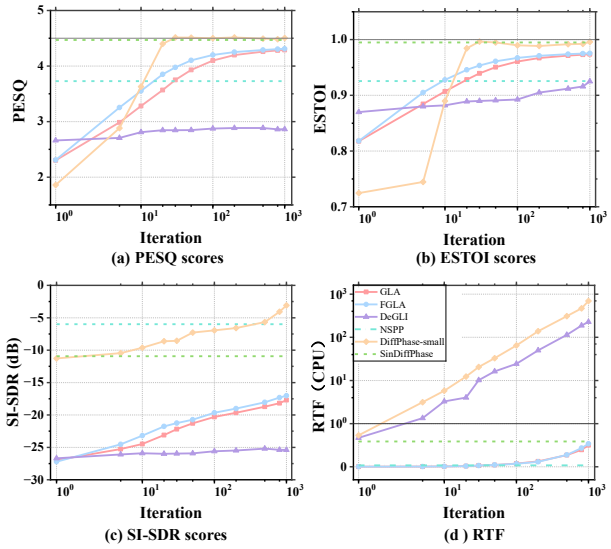

Fig. 2. Average scores for different methods across iterations (1, 5, 10, 20, 30, 50, 100, 200, 500, 800, 1000). Solid lines: methods evaluated at all iterations (GLA, FGLA, DeGLI, DiffPhase-small). Dashed lines: methods fixed at 1 iteration (NSPP, SinDiffPhase). RTF lower values indicate faster output speed.

- NSPP: It is a single-iteration model with a training batch size of 16, and 3100 training epochs.
- DiffPhase-small: This is a diffusion-based model with 1000 training discrete-time steps and n discrete testing steps, trained for 1700 epochs. Other settings are the same as those of SinDiffPhase.

The results of different iteration counts n (1, 5, 10, 20, 30, 50, 100, 200, 500, 800, 1000) are shown in Fig. 2. All iterative methods (GLA, FGLA, DeGLI, DiffPhase-small) are evaluated across all iterations. In contrast, NSPP and SinDiffPhase are fixed at 1 iteration (dashed lines) because of their single-step architecture. As shown in Fig. 2, with increasing iterations, PESQ and ESTOI of all iterative methods show a consistent upward trend until large-scale phase errors are addressed, then converge to stable values. While SI-SDR and RTF increase with the number of iterations. This discrepancy among PESQ, ESTOI, and SI-SDR can be attributed to their differing perceptual sensitivities. SI-SDR monotonically reflects even minute phase distortions without any perceptual bound, whereas PESQ and ESTOI are grounded in human auditory perception. Owing to the finite resolution of the auditory system, the latter two metrics plateau once phase errors fall below the threshold of perceptibility, rendering any further refinement inaudible and thus unreflected in the scores. Among all methods, diffusion-based models demonstrate superior phase estimation quality, with DiffPhase-small requiring high iteration counts (typically > 30) to achieve excellent performance, although its RTF makes it unsuitable for real-time applications. In contrast, SinDiffPhase surpasses multiple traditional and DNN methods by achieving PESQ and ESTOI values comparable to those of DiffPhase-small with only 1 iteration, while ensuring real-time processing capability.

TABLE I
COMPARISON OF PERFORMANCE METRICS AND PHASE-SPECIFIC LOSSES ACROSS DIFFERENT METHODS

Methods	Iterations	PESQ ↑	ESTOI ↑	SI-SDR (dB) ↑	RTF ↓	$\mathcal{L}_{\text{IP}}$ ↓	$\mathcal{L}_{\text{GD}}$ ↓	$\mathcal{L}_{\text{IAF}}$ ↓
GLA20	20	3.57	0.928	-22.94	0.015	1.57	0.79	0.79
GLA100	100	4.10	0.961	-20.82	0.074	1.57	0.58	0.58
FGLA20	20	3.85	0.946	-21.78	0.016	1.57	0.70	0.70
FGLA100	100	4.20	0.967	-19.38	0.068	1.57	0.50	0.50
DeGLI	100	2.88	0.893	-25.60	24.422	1.57	1.04	1.29
NSPP	1	3.73	0.926	-6.00	0.033	1.55	0.93	1.09
DiffPhase-small	30	4.51	0.996	-8.85	20.575	1.53	0.18	0.19
SinDiffPhase	1	4.45	0.994	-10.94	0.586	1.54	0.11	0.11

[a] ↑: Higher values indicate better performance
[b] ↓: Lower values indicate better performance

To analyze architectural differences, Table I compares several representative methods illustrated in Fig. 2. In Table I, the instantaneous phase loss ($\mathcal{L}_{\text{IP}}$), group delay loss ($\mathcal{L}_{\text{GD}}$), and instantaneous angular frequency loss ($\mathcal{L}_{\text{IAF}}$) are added to observe method performance. Due to varying speech lengths in the test set, the T-F bin-level computation is used. Regarding the loss metrics, each method shows little difference in minimizing $\mathcal{L}_{\text{IP}}$. SinDiffPhase with a single iteration demonstrates superior performance in minimizing $\mathcal{L}_{\text{GD}}$ and $\mathcal{L}_{\text{IAF}}$, surpassing all other methods, which indicates that its phase estimation is closer to the ground truth in terms of group delay in the frequency domain and instantaneous angular frequency variation in the time domain. For speech quality metrics, SinDiffPhase achieves PESQ scores comparable to DiffPhase-small and surpasses other methods, which reflects the superiority of its phase estimation in improving speech quality. SinDiffPhase's ESTOI score (0.994) is nearly indistinguishable (within 0.2%) from DiffPhase-small with 30 iterations (0.996), outperforming other methods by a clear margin, and validating that its single-step design retains high perceptual intelligibility. In SI-SDR, it outperforms most methods except NSPP and DiffPhase-small. Achieving an RTF of 0.586, SinDiffPhase is 35.1× faster than DiffPhase-small (20.575) due to its single-step iteration design (reduced from 30 iterations of DiffPhase-small to 1). As a result, SinDiffPhase resolves the long-standing trade-off between speed and quality.

Therefore, SinDiffPhase achieves optimal comprehensive performance: It can optimize group delay loss and instantaneous angular frequency loss more effectively, and it enables high-speed processing while maintaining high perceptual quality and intelligibility.

C. Ablation Studies

Ablation studies are then conducted to explore the roles of discrete-time step settings and the anti-wrapping phase loss function in the proposed SinDiffPhase. All ablated variants are modified based on SinDiffPhase, as detailed below:

- **Diff-Ran+Ran**: Randomly select discrete-time steps (ranging from 0 to 1000) for both training and testing phases, replacing the original fixed-step logic of SinDiffPhase.
- **Diff-Ran+30**: Randomly choose discrete-time steps (0 to 1000) during training, while setting testing discrete-time steps to 30, altering the step-setting strategy of SinDiffPhase.

TABLE II
ABLATION STUDY RESULTS: PERFORMANCE COMPARISON OF SINDIFFPHASE AND VARIANTS WITH MODIFIED DISCRETE-TIME STEPS OR LOSS FUNCTION

Methods	PESQ ↑	ESTOI ↑	SI-SDR (dB) ↑	RTF ↓
SinDiffPhase	4.33	0.990	-16.60	0.586
Diff-Ran+Ran	3.96	0.956	-18.77	239.720
Diff-Ran+30	4.00	0.963	-19.44	21.075
Diff-1000+30	3.84	0.947	-17.59	21.070
Diff-SL	1.86	0.686	-20.57	0.582
DiffPhase-small	4.32	0.977	-16.60	20.575

[a] ↑: Higher values indicate better performance
[b] ↓: Lower values indicate better performance

- **Diff-1000+30**: Use 1000 discrete-time steps for training and 30 for testing.
- **Diff-SL**: Replace the phase loss function in SinDiffPhase with the original score loss function.

DiffPhase-small, SinDiffPhase and variants are trained for 100 epochs, with all other settings held identical. The experimental results in Table II show that SinDiffPhase and Diff-SL have comparable generation speed, which is faster than all other methods. This confirms that setting the number of discrete-time steps to 1 breaks through the processing speed limit of conventional diffusion methods. SinDiffPhase outperforms Diff-Ran+Ran, Diff-Ran+30, and Diff-1000+30 in PESQ, ESTOI and SI-SDR, indicating that increasing the training and generation discrete-time steps with the phase loss function fails to improve phase estimation quality. Moreover, the significantly lower PESQ, ESTOI, and SI-SDR scores of Diff-SL compared to SinDiffPhase confirm that the phase loss function is more suitable for single-step training and sampling. This discrepancy may be caused by the score loss function that relies on multistep denoising, which fails to capture global phase correlation in single-step inference. SinDiffPhase achieves phase accuracy comparable to DiffPhase-small but with a 35.1× faster generation speed and significantly higher training efficiency. Therefore, combining phase loss with single-step training and sampling maintains a high-quality estimation while boosting speed.

V. CONCLUSIONS

Inspired by diffusion models, this paper proposes SinDiffPhase, which sets both the training and testing discrete-time steps to 1. This is obtained by incorporating the anti-wrapping phase loss function into the DiffPhase-small architecture. For the first time, this model achieves ultra-fast single-step diffusion for phase estimation, overcoming the speed

constraints of traditional diffusion methods. Multidimensional experiments demonstrate that SinDiffPhase enables rapid high-fidelity phase estimation. Notably, it achieves high-speed output while maintaining the phase estimation quality comparable to that of DiffPhase-small. Ablation results confirm that unifying discrete-time steps and integrating phase loss are crucial for model performance.

References

[1] K. Paliwal, K. Wójcicki, and B. Shannon, "The importance of phase in speech enhancement," *Speech Commun.*, vol. 53, no. 4, pp. 465–494, 2011.

[2] C. Zheng, H. Zhang, W. Liu, *et al.*, "Sixty years of frequency-domain monaural speech enhancement: From traditional to deep learning methods," *Trends Hear.*, vol. 27, p. 23 312 165 231 209 913, 2023.

[3] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev, "Phase retrieval with application to optical imaging: A contemporary overview," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 87–109, 2015.

[4] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase processing for single-channel speech enhancement: History and recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 55–66, 2015.

[5] P. Mowlaee, R. Saeidi, and Y. Stylianou, "Advances in phase-aware signal processing in speech communication," *Speech Commun.*, vol. 81, pp. 1–29, 2016.

[6] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 32, no. 2, pp. 236–243, 1984.

[7] N. Perraudin, P. Balazs, and P. L. Søndergaard, "A fast griffin-lim algorithm," in *Proc. WASPAA*, IEEE, 2013, pp. 1–4.

[8] G. T. Beauregard, X. Zhu, and L. Wyse, "An efficient algorithm for real-time spectrogram inversion," in *Proc. DAFx*, 2005, pp. 116–118.

[9] T. Kobayashi, T. Tanaka, K. Yatabe, and Y. Oikawa, "Acoustic application of phase reconstruction algorithms in optics," in *Proc. ICASSP*, IEEE, 2022, pp. 6212–6216.

[10] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Deep griffin–lim iteration," in *Proc. ICASSP*, IEEE, 2019, pp. 61–65.

[11] Y. Masuyama, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Deep griffin–lim iteration: Trainable iterative phase reconstruction using neural network," *IEEE J. Sel. Top. Signal Process.*, vol. 15, no. 1, pp. 37–50, 2020.

[12] Y. Zhang, M. Andreas Noack, P. Vagovic, *et al.*, "Phasegan: A deep-learning phase-retrieval approach for unpaired datasets," *Opt. Express*, vol. 29, no. 13, pp. 19 593–19 604, 2021.

[13] T. Peer, S. Welker, and T. Gerkmann, "Beyond griffin-lim: Improved iterative phase retrieval for speech," in *Proc. IWAENC*, IEEE, 2022, pp. 1–5.

[14] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *Proc. CVPR*, 2022, pp. 11 461–11 471.

[15] J.-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, "Storm: A diffusion-based stochastic regeneration model for speech enhancement and dereverberation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 31, pp. 2724–2737, 2023.

[16] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, "Speech enhancement and dereverberation with diffusion-based generative models," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 31, pp. 2351–2364, 2023.

[17] C. Luo, "Understanding diffusion models: A unified perspective," *arXiv preprint arXiv:2208.11970*, 2022.

[18] T. Peer, S. Welker, and T. Gerkmann, "Diffphase: Generative diffusion-based stft phase retrieval," in *Proc. ICASSP*, IEEE, 2023, pp. 1–5.

[19] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. ICASSP*, IEEE, vol. 2, 2001, pp. 749–752.

[20] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio Speech Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[21] Y. Ai and Z.-H. Ling, "Neural speech phase prediction based on parallel estimation architecture and anti-wrapping losses," in *Proc. ICASSP*, IEEE, 2023, pp. 1–5.

[22] G. E. Uhlenbeck and L. S. Ornstein, "On the theory of the brownian motion," *Phys. Rev.*, vol. 36, no. 5, p. 823, 1930.

[23] J. Zhang, D. Liu, E. Park, S. Zhang, and C. Xu, "Anti-exposure bias in diffusion models via prompt learning," in *Proc. ICLR*.

[24] C. Veaux, J. Yamagishi, and S. King, "The voice bank corpus: Design, collection and data analysis of a large regional accent speech database," in *Proc. O-COCOSDA/CASLRE*, IEEE, 2013, pp. 1–4.

[25] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?" In *Proc. ICASSP*, IEEE, 2019, pp. 626–630.