Speaking in Wavelet Domain: A Simple and Efficient Approach to Speed up Speech Diffusion Model

Xiangyu Zhang[1*], **Daijiao Liu**[1*], **Hexin Liu**[3], **Qiquan Zhang**[1]
Hanyu Meng[1], **Leibny Paola Garcia**[4], **Eng Siong Chng**[3], **Lina Yao**[2]
The University of New South Wales[1], Data61 CSIRO[2]
Nanyang Technological University[3], HLTCOE and Johns Hopkins University[4]

Abstract

Recently, Denoising Diffusion Probabilistic Models (DDPMs) have attained leading performances across a diverse range of generative tasks. However, in the field of speech synthesis, although DDPMs exhibit impressive performance, their long training duration and substantial inference costs hinder practical deployment. Existing approaches primarily focus on enhancing inference speed, while approaches to accelerate training—a key factor in the costs associated with adding or customizing voices—often necessitate complex modifications to the model, compromising their universal applicability. To address the aforementioned challenges, we propose an inquiry: **is it possible to enhance the training/inference speed and performance of DDPMs by modifying the speech signal itself?** In this paper, we **double** the training and inference speed of Speech DDPMs by simply redirecting the generative target to the wavelet domain. This method not only achieves comparable or superior performance to the original model in speech synthesis tasks but also demonstrates its versatility. By investigating and utilizing different wavelet bases, our approach proves effective not just in speech synthesis, but also in speech enhancement.

1 Introduction

Recently, with the advancement of deep learning, generative models have made significant progress in various fields (Karras et al., 2019; Oord et al., 2016; Yang et al., 2019). Particularly, the emergence of diffusion models has elevated the capabilities of deep generative models to a new level (Ho et al., 2020; Song et al., 2020b). In the field of speech processing, Denoising Diffusion Probabilistic Models (DDPMs) not only exhibit astonishing performance in speech synthesis (Kong et al., 2020; Jeong et al., 2021) but also demonstrate commendable results in speech enhancement (Lu et al., 2022; Yen et al., 2023). However, despite the impressive results achieved by DDPMs in the field of speech processing, the requirement to generate a guarantee of high sample quality — typically necessitating hundreds to thousands of denoising steps — results in training and inference speeds that are daunting in practical applications.

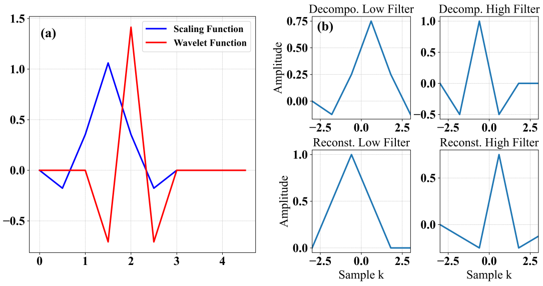


Figure 1: Wavelet of Cohen-Daubechies-Feauveau 5-tap/3-tap. (a) Scaling and wavelet functions, (b) decomposition and reconstruction filters.

Given these issues, researchers from various fields have attempted different methods to improve diffusion models. In the realm of speech processing, existing approaches have endeavored to alter the model structure to accelerate the inference speed of speech synthesis (Huang et al., 2022), while others have experimented with changing training strategies to reduce the number of inference steps required for diffusion models in speech enhancement (Lay et al., 2023). These approaches primarily focus on enhancing the inference speed of speech diffusion models. However, in the field of speech synthesis, the industry frequently requires incorporating new voices to accommodate varied requirements. Additionally, generative-based speech enhancement often demands tailoring models to distinct scenarios, which introduces practical limitations to the aforementioned methods in real-world applications. In the field of computer vision, researchers have attempted to accel-

[*] Equal contribution, Our Code Can be found in`https://github.com/Tonyyouyou/WaveD_TTS`

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 159–171
November 12-16, 2024 ©2024 Association for Computational Linguistics

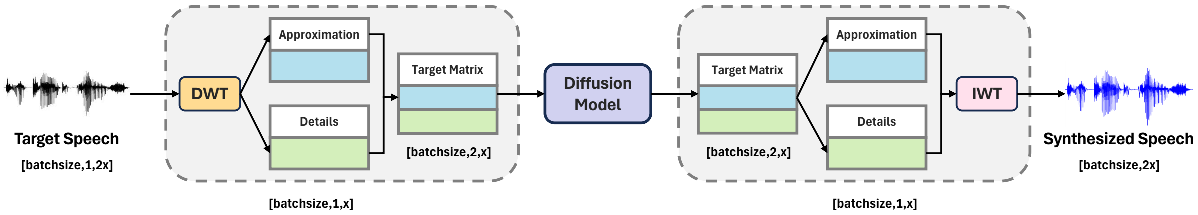


Figure 2: Overview of the Speech Wavelet Diffusion Model pipeline: First, the speech signal is decomposed into Approximation coefficients Matrix(cA) and Detail coefficients matrix(cD), the Diffusion model subsequently generates cA and cD and restores the speech signal from these matrices.

erate diffusion models using wavelets. Their efforts are mainly concentrated on score-based diffusion models (Song et al., 2020b, 2021), employing wavelets to modify the training strategy, thereby simultaneously enhancing both training and inference speeds (Guth et al., 2022). **However, there is a significant difference between audio and image signals.** Unlike the common feature sizes of 64x64 or 256x256 in images, speech signals often have large feature sizes to ensure training quality. This means that the challenges in training speech models often stem from the nature of the speech signal itself (Radford et al., 2023). Considering this, we propose a question from a different angle: can we improve the training and inference speeds of DDPMs and significantly alleviate GPU memory pressure by operating directly on the speech signal itself?

The principle of simplicity often underlies effective methods, as evidenced by tools like LoRA (Hu et al., 2021) and Word2Vec (Mikolov et al., 2013). Inspired by the successful application of latent space diffusion models (Rombach et al., 2022) and wavelets in image compression (Taubman et al., 2002), we pivot the generative aim of speech DDPMs towards the compressed speech signal in the wavelet domain. This involves decomposing the speech signal using the Discrete Wavelet Transform(DWT) into high-frequency and low-frequency components. These components are then concatenated to form a unified generative target for our model. Through this approach, the feature-length of the data is halved, which enhances the GPU's parallel processing capabilities and significantly reduces the demand for GPU memory.

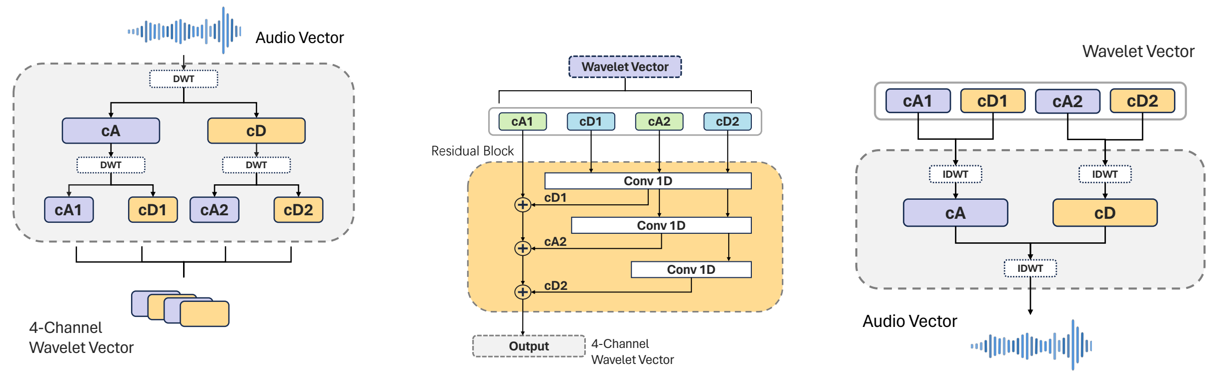
In the Further Study chapter, we have developed two additional modules: the Low Frequency Enhancer and the Multi-Level Accelerator. The former enhances low-frequency signals, allowing our method to not only double the speed compared to the original model but also achieve better performance. The latter, by integrating the Low-Frequency Enhancer with multi-level wavelet transform, further compress the speech signal. This enables an acceleration of more than five times while maintaining comparable results.

In summary, our contributions include the following:

- We designed a simple, effective, and universal method that **doubles the training and inference speed** of the original model **without altering its architecture** while maintaining comparable performance. Testing across different models and tasks not only confirmed the wide applicability and versatility of our approach but also demonstrated that the Diffusion Models can generate speech components in the wavelet domain.
- We designed two simple and easily integrable front-end modules. The first achieves **better performance than the original model while doubling the speed**. The second offers a performance comparable to the original while enabling an acceleration of more than five times.
- We offer a new perspective on accelerating and optimizing speech models by focusing on processing the signal itself rather than modifying the model, thereby charting a new course for future research.

2 Related Work

Diffusion Probabilistic Models. Diffusion probabilistic models (DMs) (Sohl-Dickstein et al., 2015; Ho et al., 2020) are a powerful and effective class of generative models, which are highly competitive in terms of sample quality, surpassing Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) to become the state-of-the-art in a variety of synthesis tasks (Dhariwal and Nichol, 2021; Liu et al., 2022). DMs comprise a forward noise diffusion process and a Markovian reverse



(a) Block of Multi-Level Discrete Wavelet Transform

(b) Multi-Level Low-Frequency Voice Enhancement Module

(c) Block of Multi-Level Inverse Discrete Wavelet Transform

Figure 3: Overview of (a) Block of Multi-Level Discrete Wavelet Transform, (b) Multi-Level Low-Frequency Voice Enhancement Module, (c) Block of Multi-Level Inverse Discrete Wavelet Transform.

diffusion process. They function by training a deep neural network to denoise content that has been corrupted with various levels of Gaussian noise. In the sampling phase, a generative Markov chain process based on Langevin dynamics (Song and Ermon, 2019) iteratively denoises from complete Gaussian noise to progressively generate the target samples. Due to their iterative nature, DMs experience a significant increase in training and sampling time when generating high-dimensional data (Song et al., 2020a).

Speech Synthesis. In recent times, a variety of neural text-to-speech (TTS) systems have been developed (Oord et al., 2016; Bińkowski et al., 2019; Valle et al., 2020; Chen et al., 2024). Initially, these systems generate intermediate representations, such as mel spectrograms or hidden representations, conditioned on textual input. This is followed by the use of a neural vocoder for the synthesis of the raw audio waveform. The pivotal role in the recent advancements of speech synthesis has been played by neural vocoders. Models like WaveFlow (Ping et al., 2020) and WaveGlow (Prenger et al., 2019) achieve training through likelihood maximization. On the other hand, models based on VAEs and GANs diverge from likelihood-centric models, often necessitating additional training losses to enhance audio fidelity. Another notable approach is the diffusion-based model (Kong et al., 2020), which stands out by synthesizing high-quality speech using a singular objective function. Our experiment will be conducted on a diffusion-based vocoder.

Speech Enhancement. Speech enhancement is a field in audio signal processing focused on improving the quality of speech signals in the presence of noise (Benesty et al., 2006). Recent advances in deep learning have significantly improved the performance of speech enhancement systems, enabling more effective noise suppression and clarity in diverse environments (Zhang et al., 2020; Sun et al., 2023; Zhang et al., 2024). In the realm of speech denoising, diffusion-based models are being effectively utilized. Lu (Lu et al., 2022) investigates the efficacy of diffusion model with noisy mel band inputs for this purpose. In a similar vein, Joan (Serrà et al., 2022) examines the application of score-based diffusion models for enhancing speech quality. Furthermore, Welker (Welker et al., 2022) proposes formulations of the diffusion process specifically designed to adapt to real audio noises, which often present non-Gaussian properties.

Speed Up Generative Speech Model. Numerous efforts have been made to expedite speech synthesis, with Fastspeech (Ren et al., 2019) and Fastspeech 2 (Ren et al., 2020) being among the most notable, both accelerating the process using transformer models. FastDiff (Huang et al., 2022), a more recent development, aims to address the slow inference speed of diffusion models in practical applications, focusing primarily on hastening inference time. In contrast, our technology is designed **not only to accelerate both training and inference but also to be easily adaptable to various speech synthesis models**.

3 Methodology

In this section, the proposed method is illustrated using the Cohen-Daubechies-Feauveau 5/3 wavelet as a case study (Le Gall and Tabatabai, 1988). We first explain how we utilize wavelet transforms for compressing and parallel processing of speech sig-

Algorithm 1 Wavelet Diffwave Training

```
for $i = 1, 2, \dots, N_{\text{iter}}$ do
    Sample $x_0 \sim q_{\text{data}}, \epsilon \sim \mathcal{N}(0, \mathbf{I})$, and
    $t \sim \text{Uniform}(\{1, \dots, T\})$
    $y_0 = DWT(x_0)$
    Take gradient step on
    $\nabla_\theta \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} y_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t)\|_2^2$
```

Algorithm 2 Wavelet Diffwave Sampling

```
Sample $ye_T \sim p_{\text{latent}} = \mathcal{N}(0, \mathbf{I})$
for $t = T, T - 1, \dots, 1$ do
    Compute $\mu_\theta(y_t, t)$ and $\sigma_\theta(y_t, t)$
    Sample $y_{t-1} \sim p_\theta(y_{t-1}|y_t) =$
        $\mathcal{N}(y_{t-1}; \mu_\theta(y_t, t), \sigma_\theta(y_t, t)^2\mathbf{I})$
$x_0 = IWT(y_0)$
return $x_0$
```

nals. Then, we delve into the specifics of accelerating speech synthesis and enhancement tasks.

3.1 Wavelet Transform and Compression

The Wavelet Transform is a key method in image compression, involving Discrete Wavelet Transform (DWT) and Inverse Discrete Wavelet Transform (IWT) to separate low-frequency (cA) and high-frequency (cD) components from signals (Sullivan, 2003). We focus on the Daubechies-Feauveau 5/3 wavelet, shown in Figure 1, a biorthogonal wavelet commonly used in lossless compression algorithms (Taubman et al., 2002). Let us define $L = \left[-\frac{1}{8}, \frac{2}{8}, \frac{6}{8}, \frac{2}{8}, -\frac{1}{8}\right]$ and $H = \left[\frac{1}{2}, 1, \frac{1}{2}\right]$ as the low-pass and high-pass filters, respectively. In the DWT Process, these filters are employed to decompose speech signals $x \in \mathbb{R}^{1\times 2x}$ into matrices $cA \in \mathbb{R}^{1\times x}$ and $cD \in \mathbb{R}^{1\times x}$. Subsequently, these matrices are concatenated to form $y \in \mathbb{R}^{2\times x}$, as depicted in the left part of Figure 2. In the IWT process, the matrix $y \in \mathbb{R}^{2\times x}$ is divided back into $cA \in \mathbb{R}^{1\times x}$ and $cD \in \mathbb{R}^{1\times x}$, which are then reconstructed into the speech signal. The details of how Wavelet compresses speech and accelerates the model can be seen in Appendix C.

3.2 Wavelet-based Speech Diffusion Scheme

3.2.1 Speech Synthesis

We evaluated our method using Diffwave (Kong et al., 2020), a well-known diffusion vocoder widely adopted in numerous TTS systems. We altered only the first layer of the one-dimensional convolutional network used for processing the input signal, ensuring that the number of channels remains constant, thereby keeping the network width unchanged in comparison with Diffwave. During the training process, the diffusion process is characterized by a fixed Markov chain transitioning from the concatenated wavelet data y_0 to the latent variable y_T. This is achieved via

$$q(y_1, \dots, y_T|y_0) = \textstyle\prod_{t=1}^{T} q(y_t|y_{t-1}), \qquad (1)$$

where $q(y_t|y_{t-1})$ is defined as a Gaussian distribution $\mathcal{N}(y_t; \sqrt{1-\beta_t}y_{t-1}, \beta_t\mathbf{I})$ and β is a small positive constant. The function $q(y_t|y_{t-1})$ introduces slight Gaussian noise into the distribution of y_{t-1}, effectively adding minimal Gaussian noise to both cA and cD.

The reverse process is characterized by a Markov chain transitioning from y_T back to y_0. This is parameterized by θ and computed via

$$p_\theta(y_0, \dots, y_{T-1}|y_T) = \textstyle\prod_{t=1}^{T} p_\theta(y_{t-1}|y_t). \qquad (2)$$

The distribution $p(y_T)$ originates from an isotropic Gaussian and is composed of two distinct components, corresponding respectively to cA and cD. The term $p_\theta(y_{t-1}|y_t)$ is parameterized by a Gaussian distribution $\mathcal{N}(y_{t-1}; \mu_\theta(y_t, t), \sigma_\theta(y_t, t)^2\mathbf{I})$. Here, μ_θ yields a $2 \times X$ matrix representing the mean values for cA and cD, while σ_θ produces two real numbers, indicating the standard deviations for cA and cD.

The training objective is to minimize the following unweighted variant of the variational lower bound (ELBO):

$$\min_\theta L(\theta) = \mathbb{E}\left\|\epsilon - \theta\left(\sqrt{\overline{\alpha}_t}y_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, t\right)\right\|^2 \qquad (3)$$

where $\overline{\alpha}_t$ is derived from the variance schedule, parameter θ denotes a neural network that outputs noise for both cA and cD. Furthermore, ϵ is represented as a $2 \times X$ matrix, encapsulating the actual noise values corresponding to both cA and cD. The detailed procedures for training and sampling are outlined in Algorithm 1 and Algorithm 2.

3.2.2 Speech Enhancement

We also evaluated our algorithm in Diffusion-based Speech Enhancement tasks, employing CDiffuSE (Lu et al., 2022) as a test case to demonstrate the effectiveness of our approach. Their diffusion forward process after wavelet processing can be formulated as

$$q_{\text{diff}}(y_t|y_0, y_n) = \mathcal{N}\left(y_t; (1-m_t)\sqrt{\overline{\alpha}_t}y_0 + m_t\sqrt{\overline{\alpha}_t}y_n, \delta_t\mathbf{I}\right). \qquad (4)$$

Algorithm 3 Wavelet CDiffuSE Sampling

```
Sample $y_T \sim \mathcal{N}(y_T, \sqrt{\bar{\alpha}_T} y_n, \delta_T \mathbf{I})$
for $t = T, T-1, \dots, 1$ do
    Compute $c_{x_t}, c_{y_t}$ and $c_{\epsilon_t}$
    Sample $y_{t-1} \sim p_\theta(y_{t-1}|y_t, y_n) = \mathcal{N}(y_{t-1}; c_{x_t} y_t + c_{y_t} y_n - c_{\epsilon_t} \epsilon_\theta(y_t, y_n, t), \delta_t \mathbf{I})$
    $x_0 = IWT(y_0)$
return $x_0$
```

Algorithm 4 Wavelet CDiffuSE Training

```
for $i = 1, 2, \dots, N_{\text{iter}}$ do
    Sample $(x_0, x_n) \sim q_{\text{data}}, \epsilon \sim \mathcal{N}(0, \mathbf{I})$,
    $y_0 = DWT(x_0), y_n = DWT(x_n)$
    $t \sim \text{Uniform}(\{1, \dots, T\})$
    $y_t = ((1-m_t)\sqrt{\bar{\alpha}_t} y_0 + m_t \sqrt{\bar{\alpha}_t} y_n) + \sqrt{\delta_t} \epsilon$
    Take gradient step on $\nabla_\theta \left\| \frac{1}{\sqrt{1-\bar{\alpha}_t}} (m_t \sqrt{\bar{\alpha}_t}(y_n - y_0) + \sqrt{\delta_t} \epsilon) - \epsilon_\theta(y_t, y_n, t) \right\|_2^2$
```

The variable m_t represents the interpolation ratio between the clean wavelet data y_0 and the noisy wavelet data y_n. This ratio initiates at $m_0 = 0$ and progressively increases to $m_t = 1$. The term $\bar{\alpha}_t$ is computed following the same methodology as employed in Diffwave, and δ_t is defined as $(1 - \alpha_t) - m_t^2 \alpha_t$. The reverse process is formulated as

$$p_\theta(y_{t-1}|y_t, y_n) = \mathcal{N}(y_{t-1}; \mu_\theta(y_t, y_n, t), \tilde{\delta}_t \mathbf{I}), \quad (5)$$

where $\mu_\theta(y_t, y_{noise}, t)$ is the mean of a linear combination of y_t and y_{noise}, being formulated as

$$\mu_\theta(y_t, y_n, t) = c_{y_t} y_t + c_{y_n} y_n - c_{\epsilon_t} \epsilon_\theta(y_t, y_n, t). \quad (6)$$

Parameters c_{y_t}, c_{y_n}, and c_{ϵ_t} are derived from the ELBO optimization. The detailed procedures for training and sampling are outlined in Algorithm 4 and Algorithm 3. The details of coefficients and ELBO optimization can be seen in Appendix B.

4 Experiments

4.1 Dataset

Speech Synthesis Our experiments were conducted using the LJSpeech dataset (Ito and Johnson, 2017), comprising 13,100 English audio clips along with their corresponding text transcripts. The total duration of the audio in this dataset is approximately 24 hours. For the purpose of objectively assessing the NISQA Speech Naturalness (Mittag et al., 2021), 1,000 samples were randomly chosen as the test dataset. Additionally, we conduct a subjective audio evaluation using a 5-point Mean Opinion Score (MOS) test, involving 30 examples per model and 20 participants.

Speech Enhancement Our experiments were conducted using the VoiceBankDEMAND dataset (Valentini-Botinhao et al., 2016). The dataset, derived from the VoiceBank corpus (Veaux et al., 2013), encompasses 30 speakers and is bifurcated into a training set with 28 speakers and a testing set with 2 speakers.The training utterances are deliberately mixed with eight real-recorded noise samples from the DEMAND database, in addition to two synthetically generated noise samples, at SNR levels of 0, 5, 10, and 15 dB. This results in a total of 11,572 training utterances.

For testing, the utterances are combined with different noise samples at SNR levels of 2.5, 7.5, 12.5, and 17.5 dB, culminating in a total of 824 testing utterances. Our algorithm was evaluated using the Perceptual Evaluation of Speech Quality (PESQ) and a deep learning evaluation approach, DNSMos (Dubey et al., 2023).

4.2 Model Architecture and Training

To ensure a fair comparison with the baseline, we adhered to the identical parameter settings utilized in both Diffwave and CDiffuSE. To more effectively validate the versatility of our method, we conducted tests on both the base and large versions of Diffwave and CDiffuSE. To explore the distinct characteristics of various wavelets, we conducted experiments using a computational base of 32 NVIDIA V100 32GB GPUs. we conducted tests with different wavelets base using 32 V100 32G, including Haar, Biorthogonal 1.1 (bior1.1), Biorthogonal 1.3 (bior1.3), Coiflets 1 (coif1) (Daubechies, 1988), Daubechies 2 (db2), and Cohen-Daubechies-Feauveau 5/3 (cdf53) (Sullivan, 2003). The details of the parameter setting can be seen in Appendix A.

4.3 Main Result

Table 1 shows the results for various wavelet bases in both Speech Enhancement and Speech Synthesis tasks. It can be observed that, across all tasks, regardless of the type of wavelet basis used, the training time, the inference time, and the required GPU memory consumption have been reduced by nearly half. In the Speech Enhancement task, when evaluated using the pseq metric, most wavelets, with the exception of the Coif1, performed comparably to the original model. The **DB2 wavelet** exhibited the best performance on both the base and large models.

Despite nearly doubling in training and inference speeds, its performance was only marginally lower than the original model, with a difference of 0.051 and 0.021, respectively. However, when we switch to using the DNSMos metric for evaluation, the scenario changes completely. When evaluating with the DNSMos metric, there is a complete shift in results. The **Coif1 wavelet** becomes the best performer. In the base model, it surpasses the original model by 0.009, and in the large model, the lead extends to 0.056. A detailed analysis will be presented in the subsequent sections.

In the task of Speech Synthesis, the results show some variations. In the base model, the Coif1 wavelet still outperforms others, even exceeding the original model by 0.004 in Speech Naturalness (SN). However, when we examine the large model, we find that although the Coif1 wavelet continues to perform well, it is the Bior1.3 wavelet that stands out as the top performer, surpassing the original model by 0.008 in terms of SN.

Through these experiments, we have demonstrated that our method can double the training and inference speeds of the speech diffusion model while achieving results that are comparable to, or even surpass, those of the original model. The consistent performance across both base and large models further validates the generalizability of our approach. The stable results on Diffwave and CDiffuSE highlight the versatility of our method across various tasks. This advancement enables the practical application of diffusion models in the field of speech, especially the accelerated training aspect, making it feasible to customize voices and perform targeted noise reduction for specific scenarios.

5 Further Study

Under the significant acceleration achieved by our method, we explore the potential for enhancing the quality of samples through wavelet transformation and further accelerating the training and sampling process of the diffusion model.

5.1 Low-frequency Speech Enhancer

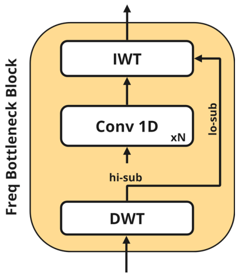


Figure 4: Overview of Frequency Bottleneck Block

In speech signals, the primary speech components are typically concentrated in the low-frequency range, while background noise tends to dominate the high-frequency spectrum (Flanagan, 2013). Therefore, to further enhance the quality of synthesized speech, we fully leverage the properties of wavelet decomposed signals. By performing Discrete Wavelet Transform (DWT) on the speech signals (Shensa et al., 1992), we obtain a 2-channel vector, consisting of detail coefficients filtered through a high-pass filter and approximation coefficients filtered through a low-pass filter. Prior to feeding into the diffusion model, this vector is processed through the Frequency Bottleneck Block as shown in Figure 4, which amplifies the low-frequency speech signals and attenuates the background noise. Since different wavelet signals emphasize various speech characteristics during DWT, we tested six types of wavelets, as shown in Table 3. The results indicate that the Haar wavelet, which focuses on signal discontinuities and rapid changes (Stanković and Falkowski, 2003), achieves superior sampling quality compared to DiffWave after processing through the Frequency Bottleneck Block module.

5.2 Multi-Level Wavelet Accelerator

To further enhance training and sampling speeds, we implemented a multi-level DWT approach, as demonstrated in Figure 3a. This method reduces the length of speech signal features to a quarter of their original size, and increases the channel count to four. Concurrently, the Frequency Bottleneck Block, designed to intensify speech signals, is expanded into the Multi-level Low-Frequency Voice Enhancement Module, which encompasses a multi-level residual block. This block is adept at progressively attenuating high-frequency components, as depicted in Figure 3b. This methodology significantly reduces both training and sampling times, with training speeds approximately five times faster than the original DiffWave and sampling speeds about three times quicker. As shown in Table 2, the Mean Opinion Score (MOS) indicates that the audio quality of the samples remains comparably high, which underscores its strong practicality.

	Speech Enhancement				Speech Synthesis			
	Base							
	PESQ ↑	DNS_MOS ↑	Training Time↓	RTF↓	MOS ↑	SN ↑	Training Time↓	RTF↓
Orignial	2.466	3.116	481.784	0.728	4.38±0.08	4.372	330.857	0.599
Haar	2.387	3.008	248.065	0.402	4.32±0.09	4.302	171.914	0.317
Bior1	2.389	3.031	248.112	0.402	4.33±0.06	4.300	172.077	0.317
Coif1	1.625	**3.125**	248.997	0.407	**4.37±0.07**	**4.376**	171.964	0.325
DB2	**2.415**	3.032	251.215	0.409	4.30±0.08	4.351	172.266	0.327
Cdf53	2.367	3.049	249.190	0.407	4.23±0.07	4.372	172.266	0.325
Bior1.3	2.302	3.027	259.831	0.413	4.32±0.09	4.331	181.914	0.342
	Large							
Original	2.514	3.140	997.688	6.387	4.41±0.08	4.395	806.158	6.055
Haar	2.463	3.127	507.813	3.366	**4.40±0.07**	4.229	408.123	3.061
Bior1	2.468	3.140	504.313	3.363	4.33±0.07	4.360	408.132	3.060
Coif1	1.660	**3.196**	511.689	3.443	4.39±0.06	4.351	412.727	3.152
DB2	**2.493**	3.125	513.384	3.445	4.35±0.07	4.374	413.210	3.144
Cdf53	2.475	3.136	512.544	3.440	4.31±0.06	4.325	412.963	3.149
Bior1.3	2.395	3.126	519.353	3.467	4.32±0.09	**4.403**	421.415	3.373
GT	–	–	–	–	4.53±0.06	–	–	–

Table 1: The table presented above displays the results for various wavelet bases in both Speech Enhancement and Speech Synthesis tasks. SN represents Speech Naturalness. GT stands for Ground Truth, referring to the raw audio from human. 'Training Time' represents the time required for training in a single epoch(seconds). 'RTF' (Real-Time Factor) is utilized as a metric to assess inference time.

Speech Synthesis (Haar Base)			
Model	**MOS**	**Training Time**	**RTF**
GT	4.53±0.06	–	–
Original	4.38±0.08	330.857	0.599
Haar2C	4.41±0.09	173.198	0.318
Haar4C	4.32±0.09	65.350	0.126

Table 2: The Table shows the result of Multi-level wavelet Accelerator, the 4C means the speech signal will be decomposed into 4 Parts.

6 Ablation Study and Analysis

6.1 Effect of Vanishing Moments, Smoothing and Complexity

From Table 1, it can be observed that Coif1 performs well on the DNSmos metric and in speech synthesis tasks, yet exhibits poor performance when evaluated using the PSEQ. The difference between DNSmos and PSEQ lies in the fact that DNSmos does not require reference audio; it is used directly to evaluate the quality of the generated speech. After listening to several sets of generated speech, we discovered that while the diffusion model using Coif1 wavelets produces clear and smooth speech, there is a significant alteration in timbre compared to the original sound. By comparing with DB2 and Haar wavelets, we can conclude that as the vanishing moment increases and complexity follows (Coif1 > DB2 > Haar), the diffusion model tends to generate clearer and smoother speech. However, once the vanishing moment reaches a certain level, the timbre of the sound is altered. This characteristic enables the selection of Coif1 wavelets in scenarios where only noise reduction is needed, or in speech synthesis tasks where timbre is of lesser concern and the emphasis is on naturalness.

6.2 Effect of Order of the Wavelet

Comparing bior1.1 with bior1.3, we observe that with an increase in the reconstruction order, both the PSEQ and DNS_MOS scores decrease. This indicates that as the reconstruction order rises, the diffusion model's ability to handle noise diminishes, although there is a slight improvement in speech synthesis tasks. We believe this is because bior1.3, compared to bior1.1, captures more high-frequency information. However, noise compared to human voice generally occupies the high-frequency range,

	Speech Enhancement				Speech Synthesis			
				Base				
	PESQ ↑	DNS_MOS ↑	Training Time↓	RTF↓	MOS ↑	SN ↑	Training Time↓	RTF↓
Orignial	2.466	3.116	481.784	0.728	4.38±0.08	4.372	330.857	0.599
Haar	**2.477**	**3.157**	249.2735	0.405	**4.41±0.09**	**4.421**	173.19	0.317
Bior1	2.429	3.118	251.908	0.405	4.36±0.08	4.353	171.490	0.318
Coif1	1.647	3.129	250.579	0.410	4.38±0.06	4.104	171.455	0.327
DB2	2.463	2.999	251.004	0.411	4.36±0.07	4.252	171.777	0.328
Cdf53	2.412	3.027	251.686	0.410	4.27±0.06	4.327	173.427	0.327
Bior1.3	2.463	3.014	258.316	0.421	4.34±0.07	4.342	182.731	0.333
				Large				
Original	2.514	3.140	997.688	6.387	4.41±0.08	4.395	806.158	6.055
Haar	2.463	3.127	507.813	3.366	4.34±0.06	4.229	408.123	3.061
Bior1	2.468	3.140	504.313	3.363	4.35±0.07	4.360	408.132	3.060
Coif1	1.660	**3.196**	511.689	3.443	4.35±0.08	4.351	412.727	3.152
DB2	**2.493**	3.125	513.384	3.445	4.37±0.07	4.374	413.210	3.144
Cdf53	2.475	3.136	512.544	3.440	**4.43±0.09**	4.325	412.963	3.149
Bior1.3	2.395	3.126	522.733	3.483	4.38±0.06	**4.403**	422.326	3.342

Table 3: The table presented above displays the results for various wavelet bases in both Speech Enhancement and Speech Synthesis tasks. SN represents Speech Naturalness. 'Training Time' represents the time required for training in a single epoch(seconds). 'RTF' (Real-Time Factor) is utilized as a metric to assess inference time.

which explains why bior1.3 performs less effectively than bior1.1 in speech enhancement tasks.

Comparing Haar (DB1) with DB2, we find that when the reconstruction order remains the same, an increase in the decomposition order enhances the performance of the wavelet speech diffusion model, especially in terms of stability and superior performance in speech enhancement. It effectively removes noise while maintaining the timbre without significant changes. In speech synthesis tasks, DB2 also shows improvement over Haar, which we attribute to the increased complexity of the wavelet.

6.3 Relationship between Wavelet base and Training/Inference Speed

From Table 1, it is evident that regardless of the wavelet used, both training and inference speeds are nearly doubled compared to the original model. The table indicates that when wavelets are applied to the diffusion model, Haar and bior1.1 exhibit similar speeds. The differences in speed between Coif1, DB2, and cdf53 are minimal, with bior1.3 being the slowest. We discovered that their speeds do not strictly correlate with their computational complexity. Our analysis suggests that the longer filter length of Bior1.3 in implementation, combined with the inherently long nature of speech signals, results in increased computational overhead.

6.4 Effect of Frequency Enhancer

After incorporating the Frequency Enhancer, most wavelet speech diffusion models showed an improvement in performance. In speech enhancement tasks, Haar, bior1.3, and cdf53 wavelets demonstrated significant improvements. Meanwhile, the training and inference speeds, compared to the wavelet diffusion model without the Frequency Enhancer, remained virtually unchanged, falling within the margin of error. Haar and Coif1 wavelets diffusion model even outperformed the original model, indicating that by simply adding a small pre-processing module, we can surpass the performance of the original model while significantly increasing training and inference speeds. However, we believe that the reasons for the performance enhancement offered by these three wavelets are not the same. For the Haar wavelet, its ability to capture discontinuities and abrupt changes in signals makes it particularly effective at handling non-stationary signals like speech. The Frequency Enhancer further amplifies this capability. Bior1.3, due to its enhanced ability to capture high-frequency signals, sees a reduction in

Model on VCTK dataset	PESQ	SN	RTF
ori base	4.2179	3.1165	0.9072
haar base	4.2069	3.1209	0.3957
bior1.1 base	4.0828	3.1473	0.4077
bior1.3 base	4.0658	3.1059	0.3987
coif1 base	4.2025	2.9393	0.4031
cdf53 base	4.1089	3.1937	0.3843
db2 base	4.1634	2.9744	0.4034
haar base*	4.2323	3.0138	0.4147
bior1.1 base*	4.2083	3.0415	0.3943
bior1.3 base*	4.1921	3.0551	0.3995
coif1 base*	4.1824	3.0406	0.4034
cdf53 base*	4.0939	3.2039	0.3949
db2 base*	4.1601	3.0479	0.4053

Table 4: Low-frequency Speech Enhancer results on VCTK dataset. RTF (Real-Time Factor) is utilized as a metric to assess inference time. SN denotes Speech Naturalness, * denotes results from Low-frequency Speech Enhancer

noise after processing with the Frequency Enhancer. Therefore, its performance improves compared to when the Frequency Enhancer is not used. For the cdf53 wavelet, it is capable of compressing signals with minimal loss. After being enhanced by the Frequency Enhancer, high-frequency noise is effectively removed, while low-frequency signals are well preserved. This lossless property is better demonstrated in the field of speech synthesis, where, after enhancement by the Frequency Enhancer, the performance slightly exceeds that of the original model in MOS tests. For detailed data, please refer to table 3.

6.5 Effect of Multi-Level Wavelet Accelerator

To further explore the potential for acceleration, we conducted tests in the field of speech synthesis using the Haar wavelet, which demonstrated the most stable performance. The results of the experiment are shown in Table 2. It can be observed that when the speech signal is split into quarters of its original length, both training and inference speeds increase by more than fivefold. However, unlike the results of splitting just once (as shown in the second row of Table 2, corresponding to the second row of Table 3), which were better than the original model, the results after splitting four times, even with the Frequency Enhancer, exhibited a notable decline in MOS values. We believe this is due to information loss caused by excessive compression. However, the substantial increase in speed still makes this method worth considering for scenarios where ultra-clear audio is not required.

6.6 Performance on Multi-Speaker Dataset

In response to concerns regarding the generalizability of our method, we conducted additional experiments using the VCTK dataset (Oord et al., 2016), applying all the wavelets tested in our original study. To further strengthen our findings, we also evaluated the performance of our low-frequency speech enhancer, which forms part of our ongoing research efforts, on the same dataset. The results, presented in Table 4, demonstrate that our approach maintains consistent performance across different datasets.

7 Conclusion

In this paper, we have enhanced the speech diffusion model by transitioning its generation target to the wavelet domain, thereby doubling the model's training and inference speeds. We offer a new perspective on accelerating speech models by focusing on processing the signal itself rather than modifying the model. Our approach has demonstrated model versatility and task adaptability across both speech enhancement and synthesis. Through our research, we found that the Coif1 wavelet is an excellent choice for scenarios requiring noise reduction without the need to preserve timbre, while the DB2 wavelet is preferable when changes in timbre must be considered. For speech synthesis tasks, the Haar wavelet offers simplicity and effectiveness, whereas the cdf53 wavelet excels at preserving information to the greatest extent. Additionally, We designed two simple and easily integrable front-end modules. The first achieves better performance than the original model while doubling the speed. The second offers a performance comparable to the original while enabling an acceleration of more than five times.

limitations

In this study, speed tests were conducted on a large-scale cluster, subject to the hardware variability inherent in the cluster (despite all GPUs being V100s, they may not be identical), which could introduce some timing inaccuracies. However, considering that the training and inference times for most wavelet-utilizing diffusion models do not significantly differ, we believe these discrepancies can be disregarded. This does not detract from our contribution of accelerating the speech diffusion model by a factor of two.

Ethics Statement

Our proposed model diminishes the necessity for high-quality speech synthesis, potentially affecting employment opportunities for individuals in related sectors, such as broadcasters and radio hosts. By lowering the training costs, our approach may impact a broader audience.

Acknowledgement

This research is supported by the RIE2025 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) (Award I2301E0026), administered by A*STAR, as well as supported by Alibaba Group and NTU Singapore.

References

Jacob Benesty, Shoji Makino, and Jingdong Chen. 2006. *Speech enhancement*. Springer Science & Business Media.

Mikołaj Bińkowski, Jeff Donahue, Sander Dieleman, Aidan Clark, Erich Elsen, Norman Casagrande, Luis C Cobo, and Karen Simonyan. 2019. High fidelity speech synthesis with adversarial networks. In *International Conference on Learning Representations*.

Chen Chen, Yuchen Hu, Wen Wu, Helin Wang, Eng Siong Chng, and Chao Zhang. 2024. Enhancing zero-shot text-to-speech synthesis with human feedback. *arXiv preprint arXiv:2406.00654*.

Ingrid Daubechies. 1988. Orthonormal bases of compactly supported wavelets. *Communications on pure and applied mathematics*, 41(7):909–996.

Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794.

Harishchandra Dubey, Ashkan Aazami, Vishak Gopal, Babak Naderi, Sebastian Braun, Ross Cutler, Hannes Gamper, Mehrsa Golestaneh, and Robert Aichner. 2023. Icassp 2023 deep noise suppression challenge. In *ICASSP*.

James L Flanagan. 2013. *Speech analysis synthesis and perception*, volume 3. Springer Science & Business Media.

Florentin Guth, Simon Coste, Valentin De Bortoli, and Stephane Mallat. 2022. Wavelet score-based generative modeling. *Advances in Neural Information Processing Systems*, 35:478–491.

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

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

R Huang, MWY Lam, J Wang, D Su, D Yu, Y Ren, and Z Zhao. 2022. Fastdiff: A fast conditional diffusion model for high-quality speech synthesis. In *IJCAI International Joint Conference on Artificial Intelligence*, pages 4157–4163. IJCAI: International Joint Conferences on Artificial Intelligence Organization.

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

Myeonghun Jeong, Hyeongju Kim, Sung Jun Cheon, Byoung Jin Choi, and Nam Soo Kim. 2021. Difftts: A denoising diffusion model for text-to-speech. *arXiv preprint arXiv:2104.01409*.

Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2020. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*.

Bunlong Lay, Jean-Marie Lemercier, Julius Richter, and Timo Gerkmann. 2023. Single and few-step diffusion for generative speech enhancement. *arXiv preprint arXiv:2309.09677*.

Didier Le Gall and Ali Tabatabai. 1988. Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques. In *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, pages 761–764. IEEE.

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

Yen-Ju Lu, Zhong-Qiu Wang, Shinji Watanabe, Alexander Richard, Cheng Yu, and Yu Tsao. 2022. Conditional diffusion probabilistic model for speech enhancement. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7402–7406. IEEE.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Gabriel Mittag, Babak Naderi, Assmaa Chehadi, and Sebastian Möller. 2021. Nisqa: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets. *arXiv preprint arXiv:2104.09494*.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Wei Ping, Kainan Peng, Kexin Zhao, and Zhao Song. 2020. Waveflow: A compact flow-based model for raw audio. In *International Conference on Machine Learning*, pages 7706–7716. PMLR.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. 2019. Waveglow: A flow-based generative network for speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3617–3621. IEEE.

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.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2020. Fastspeech 2: Fast and high-quality end-to-end text to speech. In *International Conference on Learning Representations*.

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

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Joan Serrà, Santiago Pascual, Jordi Pons, R Oguz Araz, and Davide Scaini. 2022. Universal speech enhancement with score-based diffusion. *arXiv preprint arXiv:2206.03065*.

Mark J Shensa et al. 1992. The discrete wavelet transform: wedding the a trous and mallat algorithms. *IEEE Transactions on signal processing*, 40(10):2464–2482.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020a. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. 2021. Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems*, 34:1415–1428.

Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020b. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.

Radomir S Stanković and Bogdan J Falkowski. 2003. The haar wavelet transform: its status and achievements. *Computers & Electrical Engineering*, 29(1):25–44.

Gary Sullivan. 2003. General characteristics and design considerations for temporal subband video coding. *ITU-T VCEG, document VCEG-U06, Hawaii, USA*.

Siyu Sun, Jian Jin, Zhe Han, Xianjun Xia, Li Chen, Yijian Xiao, Piao Ding, Shenyi Song, Roberto Togneri, and Haijian Zhang. 2023. A lightweight fourier convolutional attention encoder for multi-channel speech enhancement. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

David S Taubman, Michael W Marcellin, and Majid Rabbani. 2002. Jpeg2000: Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, 11(2):286–287.

Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi. 2016. Investigating rnn-based speech enhancement methods for noise-robust text-to-speech. In *SSW*, pages 146–152.

Rafael Valle, Kevin J Shih, Ryan Prenger, and Bryan Catanzaro. 2020. Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis. In *International Conference on Learning Representations*.

Christophe Veaux, Junichi Yamagishi, and Simon King. 2013. The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. In *2013 international conference oriental COCOSDA held jointly with 2013 conference on Asian spoken language research and evaluation (O-COCOSDA/CASLRE)*, pages 1–4. IEEE.

Simon Welker, Julius Richter, and Timo Gerkmann. 2022. Speech enhancement with score-based generative models in the complex stft domain. *arXiv preprint arXiv:2203.17004*.

Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. 2019. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4541–4550.

Hao Yen, François G Germain, Gordon Wichern, and Jonathan Le Roux. 2023. Cold diffusion for speech enhancement. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Qiquan Zhang, Aaron Nicolson, Mingjiang Wang, Kuldip K Paliwal, and Chenxu Wang. 2020. Deepmmse: A deep learning approach to mmse-based noise power spectral density estimation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1404–1415.

Xiangyu Zhang, Qiquan Zhang, Hexin Liu, Tianyi Xiao, Xinyuan Qian, Beena Ahmed, Eliathamby Ambikairajah, Haizhou Li, and Julien Epps. 2024. Mamba in speech: Towards an alternative to self-attention. *arXiv preprint arXiv:2405.12609*.

A Details of Experiment Setup

Diffwave offers two configurations: base and large. In the base version, the model comprises 30 residual layers, a kernel size of 3, and a dilation cycle of [1, 2, ..., 512]. It utilizes 50 diffusion steps and a residual channel count of 64. The large version maintains all parameters identical to the base, except for an increase to 128 residual channels and 200 diffusion steps. All models employed the Adam optimizer, with a batch size of 16 and a learning rate of 2×10^{-4}.We trained each DiffWave model for a total of 1 million steps.

We conducted evaluations on two versions of CDiffuSE: base and large. The base CDiffuSE model employs 50 diffusion steps, while the large CDiffuSE model uses 200 diffusion steps. Batch sizes differ, with the base CDiffuSE set to 16 and the large CDiffuSE set to 15. Both the base and large CDiffuSE models were trained for 300,000 iterations, following an early stopping scheme.

B Details of CDiffuSE

The CDiffuSE is trying to optimize the likelihood by ELBO condition for the conditional diffusion process. we further extend it to the Wavelet Latent domain.

$$\begin{aligned} ELBO = & -\mathbb{E}_q \left(D_{KL}(q_{\text{cdiff}}(\mathbf{y}_T|\mathbf{y}_0, y_n) \parallel p_{\text{latent}}(\mathbf{y}_T|y_n)) \right) \\ & + \sum_{t=2}^{T} D_{KL}(q_{\text{diff}}(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{y}_0, y_n) \parallel p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t, y_n)) \\ & - \log p_\theta(\mathbf{y}_0|\mathbf{y}_1, y_n). \end{aligned} \tag{7}$$

Parameters c_{y_t}, c_{y_n}, and c_{ϵ_t} be derived as:

$$\begin{aligned} c_{yt} &= \frac{1-m_t}{1-m_{t-1}} \frac{\delta_{t-1}}{\delta_t} \sqrt{\alpha_t} + (1-m_{t-1}) \frac{\delta_{t|t-1}}{\delta_t} \frac{1}{\sqrt{\alpha_t}}, \\ c_{yn} &= \frac{(m_{t-1}\delta_t - m_t(1-m_t)\alpha_t\delta_{t-1})\sqrt{\hat{\alpha}_{t-1}}}{1-m_{t-1}\delta_t}, \\ c_{\epsilon_t} &= \frac{(1-m_{t-1})}{\delta_t} \frac{\delta_{t|t-1}\sqrt{1-\hat{\alpha}_t}}{\sqrt{\alpha_t}}. \end{aligned} \tag{8}$$

Where δ_t variance term, all other parameters have been mentioned in main section.

C Details of Wavelet Diffusion Accelerator

C.1 How Wavelets Accelerate Diffusion models

In §3.1, we detailed the application of Discrete Wavelet Transform (DWT) and Inverse Discrete Wavelet Transform (IWT) in processing audio signals, highlighting how these techniques compress the audio signal features during the diffusion process. This section elaborates on the principles behind the acceleration offered by the Wavelet Diffusion Accelerator.

To facilitate training acceleration, the diffusion model shifts its focus from generating complete audio signals with extensive features to producing compressed speech signals in wavelet domain. In line with this shift, DWT is employed to process the raw audio signal $g\,(n) \in \mathbb{R}^{1\times 2x}$, where n denotes the sample index, through two complementary filters. Specifically, a low-pass filter ϕ extracts the low-frequency components $\mathit{\Psi}_{low} \in \mathbb{R}^{1\times 2x}$:

$$\mathit{\Psi}_{low}\,(n) = \sum_{k=-\infty}^{+\infty} g\,(k)\,\phi\,(2n-k). \tag{9}$$

And a high-pass filter ψ is utilized to extract the high-frequency portion $\mathit{\Psi}_{high} \in \mathbb{R}^{1\times 2x}$:

$$\mathit{\Psi}_{high}\,(n) = \sum_{k=-\infty}^{+\infty} g\,(k)\,\psi\,(2n-k). \tag{10}$$

To further reduce the size of the features and emphasize the signal's essential characteristics, downsampling is applied to both parts of the signal, resulting in the approximation coefficients cA and the detail coefficients cD:

$$cA = \mathit{\Psi}_{low} \downarrow 2, \tag{11}$$

$$cD = \mathit{\Psi}_{high} \downarrow 2. \tag{12}$$

At this stage, the signal $g(n) \in \mathbb{R}^{1\times 2x}$ is compressed into $h(n) \in \mathbb{R}^{2\times x}$, wherein h embodies a two-channel structure, each channel containing features of halved length.

This change significantly contributes to reducing the computational time required for training the diffusion model. To further demonstrate, we exemplify with the computational changes in the diffusion model's first convolutional layer. Assuming the output channel count is C_{out}, the kernel size is K, and the output length L_{out} remains unchanged from the input length. The formula for calculating Multiply-Accumulate Operations (MACs) per channel is:

$$MAC_{each} = K \times C_{out} \times L_{out}. \tag{13}$$

Hence, for each channel, with $h(n)$ as the input, the computational load in the first convolutional layer is halved:

$$MAC_{h(n)} = K \times C_{out} \times x = \frac{1}{2} MAC_{g(n)}. \tag{14}$$

Given the GPU's optimization for parallel computing, the increase in the number of channels does not lead to a linear increase in computational time. From experimental results, both training and sampling times of the diffusion model have a significant reduction.

C.2 Wavelets for Diffusion Acceleration: Why Not FFT

While wavelet and Fourier transforms both serve as essential tools in signal processing and share similarities in handling time and frequency domain information, this section explores why Fast Fourier Transform (FFT) is not applicable for accelerating diffusion models. This is determined by the inherent nature of the Fourier transform. Assuming $f(t)$ is the representation of the signal in the time domain and $\hat{f}(\omega)$ is its representation in the frequency domain, where t stands for time and ω for frequency, then the CFT can be described as:

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t)\, e^{-i\omega t} dt. \tag{15}$$

The Fourier transform fits the entire signal $f(t)$ with a series of sine and cosine functions, converting it into frequency domain information $\hat{f}(\omega)$. As a result, the signal is stripped of time information following this transformation. However, conventional input audio signals $f(t)$ display traits where local frequency domain features shift in response to variations in short-time segments of the time domain signal, like abrupt transitions or displacements. This lack of capability to concurrently analyze local time and frequency domain information makes the Fourier transform insufficient for accurately recreating the original audio in generative models.

In contrast, for the wavelet transform, assuming $\psi(t)$ as a basic wavelet function, let:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-b}{a}\right). \tag{16}$$

where $a, b \in \mathbb{R}$, $a \neq 0$, and the function $\psi_{a,b}(t)$ is called a continuous wavelet, generated from the mother wavelet $\psi(t)$ and dependent on parameters a and b. Therefore, the continuous wavelet transform can be written as:

$$\hat{f}(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{+\infty} f(t)\, \overline{\psi\left(\frac{t-b}{a}\right)} dt. \tag{17}$$

At this juncture, the wavelet transform converts a univariate time-domain signal $f(t)$ into a bivariate function $\hat{f}(a,b)$ encompassing both time and frequency domain information. It enables targeted analysis of local frequency domain characteristics corresponding to specific time domain segments, making it particularly well-suited for handling common non-stationary audio signals.

Besides, the wavelet transform's capability for time-frequency localization analysis ensures that downsampling and compressing cA and cD does not result in significant information loss. On the contrary, based on the Discrete Fourier Transform, FFT struggles with signal compression for diffusion acceleration due to its local frequency domain transformations affecting characteristics across the entire time domain.