NOISE LEVEL LIMITED SUB-MODELING FOR DIFFUSION PROBABILISTIC VOCODERS

Takuma Okamoto[1], *Tomoki Toda*[2,1], *Yoshinori Shiga*[1], *and Hisashi Kawai*[1]

[1]National Institute of Information and Communications Technology, Japan
[2]Information Technology Center, Nagoya University, Japan

ABSTRACT

Although diffusion probabilistic vocoders WaveGrad and DiffWave can realize real-time high-fidelity speech synthesis with a simple loss function in training, all noise components with over the full range of noise levels are predicted by one model in all iterations. This paper proposes a simple but effective noise level-limited sub-modeling framework for diffusion probabilistic vocoders Sub-WaveGrad and Sub-DiffWave. In the proposed method, DiffWave conditioned on a continuous noise level like WaveGrad, and spectral enhancement post-filtering are also provided. The proposed Sub-WaveGrad and Sub-DiffWave models are realized using 10 sub-models. These models are separately trained with different noise level limits, and only necessary sub-models are used according to the noise schedule during inference. The results of experiments using a Japanese female speech corpus indicate that both the proposed Sub-WaveGrad and Sub-DiffWave outperform vanilla WaveGrad and DiffWave in terms of the model accuracy and synthesis quality while retaining the inference speed.

Index Terms— Speech synthesis, diffusion probabilistic vocoder, WaveGrad, DiffWave, sub-modeling

1. INTRODUCTION

Given the success of WaveNet [1], neural network-based waveform generative models are investigated in speech synthesis. Moreover, Tacotron 2 [2] can realize end-to-end neural text-to-speech (TTS) for English with the same quality as human natural speech when combined with the WaveNet vocoder [3], which synthesizes speech waveforms from input acoustic features. Although the inference of a WaveNet vocoder is slow because of its autoregressive structure, many types of real-time neural vocoders based on both autoregressive and non-autoregressive structures have been investigated.

In contrast to real-time autoregressive neural vocoders such as WaveRNN [4], LPCNet [5], and FeatherWave [6], non-autoregressive models, which simultaneously synthesize all speech waveform samples, can be easily implemented as real-time neural vocoders, and many models have been investigated. Non-autoregressive neural vocoders are broadly categorized into two types. The first type consists of flow-based approaches [7] such as Parallel WaveNet [8, 9], WaveGlow [10], FloWaveNet [11], WaveVAE [12], Waveflow [13], and WG-WaveNet [14]. The other type comprises generative adversarial network (GAN)-based models [15] such as MelGAN [16], Parallel WaveGAN (PWG) [17], GAN-TTS [18, 19], VocGAN [20], HiFi-GAN [21], and Multi-band MelGAN [22]. Additionally, a signal-processing-based method, the neural source-filter [23], has been presented. However, GAN-based models must train discriminators not used in the inference, and synthesis quality highly depends on their accuracy. Additionally, most such methods [8, 9, 12, 14, 16–18, 20, 21, 23] introduce multiple loss functions with weighting parameters, complicating training. Although other flow-based methods [10, 11, 13] can be trained with simple loss functions in the time domain, the network structures must be bijections.

As promising approaches to non-autoregressive neural vocoders, WaveGrad [24] and DiffWave [25] were recently proposed based on denoising score matching [26] and diffusion probabilistic models [27]. In these models, the added noise components are predicted from mixtures of speech waveforms and Gaussian white noise with weighting factors (noise levels) as the diffusion process in training. In inference, input Gaussian white noise is iteratively converted into a speech waveform by the denoising process based on Langevin dynamics [28]. In contrast to conventional non-autoregressive models, these diffusion probabilistic vocoders can be trained with a simple loss function in the time domain without a bijective structure while realizing high-fidelity synthesis. However, the inference speed is slower than that of other neural vocoders, with a real-time factor (RTF) of less than 0.1 [24, 25].

In diffusion probabilistic vocoders, all noise components over the full range of noise levels are predicted by one model in all iterations. However, the speech signal components are dominant when noise levels are relatively low whereas the noise components are dominant when noise levels are relatively high, and they are quite different situations. Therefore, by training different sub-models for different noise level ranges, the model accuracy and synthesis quality should improve because each sub-model can concentrate on predicting specific noise components within a limited range of levels.

Motivated by the above, this paper proposes a simple but effective noise level-limited sub-modeling framework for diffusion probabilistic vocoders Sub-WaveGrad and Sub-DiffWave. As a part of the proposed method, a DiffWave conditioned on continuous noise levels like WaveGrad [24] and spectral enhancement post-filtering for diffusion probabilistic vocoders are also proposed. In this initial investigation, the proposed Sub-WaveGrad and Sub-DiffWave models were implemented using 10 sub-models with the vanilla model structures. These sub-models were separately trained with different ranges of noise levels. Only the necessary sub-models were used according to the noise schedule during inference. Although the total model size of the proposed method increases with the number of sub-models, the inference speed remains the same. The experimental results presented in Section 4 suggest that both the proposed Sub-WaveGrad and Sub-DiffWave models successfully improve the model accuracy and synthesis quality compared with vanilla models while maintaining the synthesis speed.

2. DIFFUSION PROBABILISTIC VOCODERS

In diffusion probabilistic vocoders, a gradually increasing noise schedule $\beta_1, \beta_2, \cdots, \beta_N$, where N is the number of iterations, plays an important role. In training, network model ϵ_θ conditioned on acoustic features h is trained to predict added Gaussian white

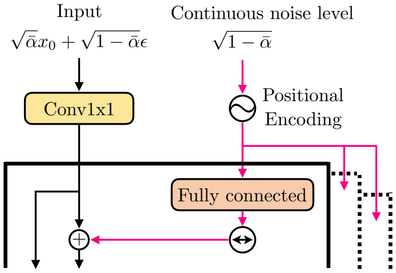


Fig. 1. Part of the network structure of the proposed DiffWave model conditioned on continuous noise level $\sqrt{1-\bar{\alpha}}$.

noise ϵ from the mixture of speech waveform x_0 and noise ϵ with a weighting factor $\bar{\alpha}_n$, where θ denotes the model parameters, $\alpha_n = 1 - \beta_n$ and $\bar{\alpha_n} = \prod_{s=1}^{n} \alpha_s$. To distinguish each noise level at each iteration, the network is also conditioned on $\sqrt{\bar{\alpha}_n}$ and n in WaveGrad and DiffWave, respectively. Additionally, to adopt continuous noise levels, a continuous noise level of $\sqrt{\bar{\alpha}}$ uniformly sampled between $\sqrt{\bar{\alpha}_n}$ and $\sqrt{\bar{\alpha}_{n-1}}$ is also conditioned on the model in WaveGrad [24]. Therefore, the loss function is simply defined in the time domain as follows:[1]

$$\mathbb{E}_{\epsilon,c}\left[\left\|\epsilon-\epsilon_\theta\left(\sqrt{\bar{\alpha}}x_0+\sqrt{1-\bar{\alpha}}\epsilon,h,c\right)\right\|_2^2\right], \qquad (1)$$

where $c = \sqrt{\bar{\alpha}}$ in WaveGrad and $c = n$ in DiffWave, respectively. In the inference, input Gaussian white noise $x_N \sim \mathcal{N}(0, I)$ is iteratively converted into a speech waveform by the denoising process based on Langevin dynamics [28] with $n = N \rightarrow 1$ as follows:

$$x_{n-1}=\frac{1}{\sqrt{\alpha_n}}\left(x_n-\frac{1-\alpha_n}{\sqrt{1-\bar{\alpha}_n}}\epsilon_\theta\left(x_n,h,c\right)\right)+\sigma_n z, \qquad (2)$$

where $\sigma_n = \sqrt{\beta_n(1-\bar{\alpha}_{n-1})/(1-\bar{\alpha}_n)}$, $z \sim \mathcal{N}(0, I)$ for $n > 1$, and $z = 0$ for $n = 1$. Although WaveGrad and DiffWave introduce the same training and synthesis strategies as Eqs. (1) and (2), the network structures differ. WaveGrad is constructed from multiple upsampling and downsampling blocks used in GAN-TTS [18], whereas DiffWave introduces WaveNet-based non-causal dilated convolution layers widely used in other non-autoregressive models.

In WaveGrad and DiffWave, only one model is used for all iterations and is trained to predict all noise components over the full range of noise levels by Eq. (1). However, the noise prediction accuracy and synthesis quality of these models should improve if the noise levels are limited, as described in Section 1.

3. PROPOSED NOISE LEVEL-LIMITED SUB-MODELING

3.1. DiffWave conditioned on continuous noise levels

Before noise level-limited sub-modeling can be proposed, DiffWave must be modified to be conditioned on continuous noise levels, like WaveGrad [24]. The adoption of continuous noise levels is important because it enables an arbitrary noise schedule to be used during inference. As a result, the synthesis quality can be improved, and fast and high-fidelity synthesis within a few iterations can be realized by an optimal noise schedule [24]. As described in Section 3.2, the proposed approach divides $\sqrt{1-\bar{\alpha}_n}$ into 10 equal parts for 10 sub-models instead of $\sqrt{\bar{\alpha}_n}$. Therefore, the noise level is defined in this study as $\sqrt{1-\bar{\alpha}}$, and DiffWave is conditioned on $c = \sqrt{1-\bar{\alpha}}$, although $\sqrt{\bar{\alpha}}$ is defined as the noise level in [24]. A part of the network structure of the proposed DiffWave conditioned on a continuous noise level $\sqrt{1-\bar{\alpha}}$ is depicted in Fig. 1, where the positional encoding used in WaveGrad is also introduced and connected to each dilated convolution layer; the remaining structure is the same as that of vanilla DiffWave. As Table 1 shows, the number of parameters of the proposed DiffWave is about 54 % of that of vanilla DiffWave conditioned on n because of the simpler structure.

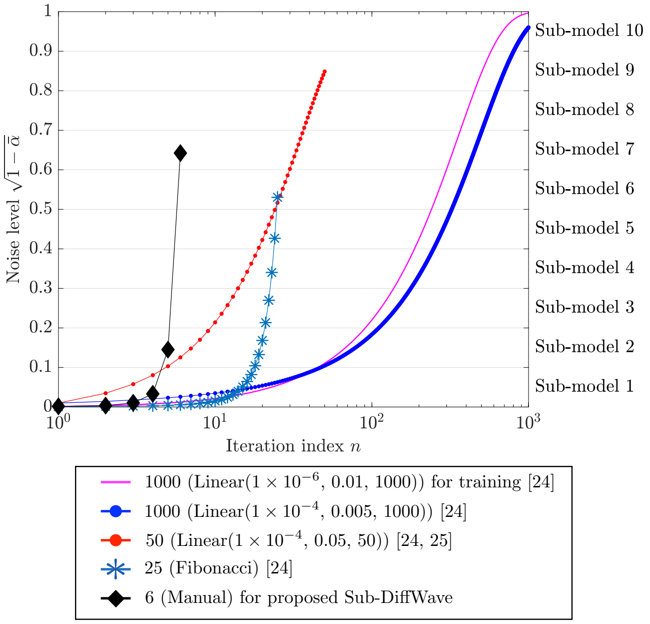


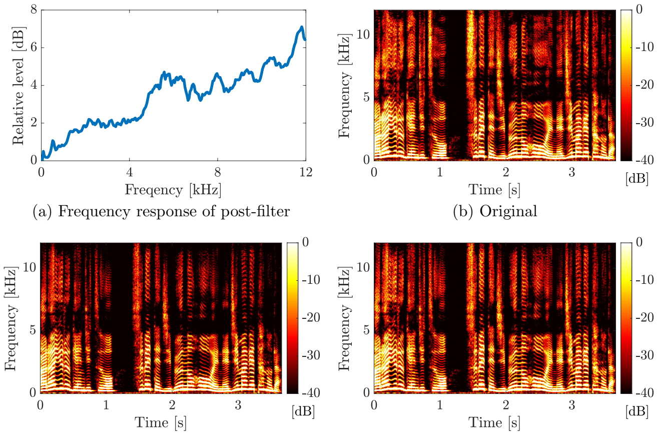
Fig. 2. Relationship between different noise schedules and the 10 proposed sub-models.

3.2. Noise level-limited sub-modeling

Here, a noise level-limited sub-modeling framework for WaveGrad and DiffWave is proposed to obtain the vocoders Sub-WaveGrad and Sub-DiffWave. First, β_1 to β_{1000} are scheduled using Linear(1×10^{-6}, 0.01, 1000), as in [24], and $\sqrt{1-\bar{\alpha}_n}$ is divided into 10 equal parts for the 10 sub-models, as shown in Fig. 2.[2] Next, each sub-model is trained conditioned on each continuous limited noise level. As described in Section 3.1, Sub-WaveGrad and Sub-DiffWave are also conditioned on $c = \sqrt{1-\bar{\alpha}}$ instead of $\sqrt{\bar{\alpha}}$. In inference, a noise schedule is set and only the necessary sub-models are used. Using the proposed sub-modeling, each sub-model is able to predict specific noise components within a limited range of noise levels. This should improve the noise prediction accuracy and speech waveform synthesis quality without reducing inference speed, although the total model size accordingly increases.

[1] Although L1 loss is introduced in WaveGrad for better stability [24], this paper introduces the mean square error (MSE) loss for WaveGrad as DiffWave because WaveGrad with the MSE loss can also be successfully trained with gradient clipping. Moreover, WaveGrad can be directly compared with DiffWave using the same loss function.

[2] Initially, $\sqrt{\bar{\alpha}_n}$ was divided into 10 equal parts for 10 sub-models according to [24]. However, it was not effective because $\sqrt{\bar{\alpha}_n}$ changes little when n is relatively small compared with $\sqrt{1-\bar{\alpha}_n}$, and most iterations are assigned to sub-model 1 during inference when N is relatively small.



(a) Frequency response of post-filter

(b) Original

(c) Synthesized by sub-DiffWave

(d) Post-filtered

Fig. 3. (a) Frequency response of proposed spectral enhancement post-filter; (b)– (d) spectrograms of a waveform in the test set for the proposed Sub-DiffWave with six iterations.

Table 1. Number of model parameters (#param), real-time factor (RTF) of the inference of the non-autoregressive neural vocoders used in the experiments, total averaged loss (TAL) of the test set scores for the diffusion probabilistic vocoders for $n = N$ to 1, and first 20 % average loss (AL 20 %) of the test set scores for $n = N$ to $0.8N + 1$, as calculated in Eq. (1). PWG: Parallel WaveGAN, (d): discrete noise condition. "-n" indicates the number of iterations.

Model	#param	RTF	TAL	AL 20 %
WaveGlow [10]	263M	0.16	-	-
PWG [17]	1.35M	0.015	-	-
WaveGrad-1000 [24]	15.8M	8.50	0.0028	0.0003
Sub-WaveGrad-1000	158M	8.50	0.0022	0.0003
WaveGrad(d)-50 [24]	15.8M	0.40	0.0058	0.0007
WaveGrad-50 [24]	15.8M	0.42	0.0063	0.0007
Sub-WaveGrad-50	142M	0.42	0.0050	0.0006
WaveGrad-25 [24]	15.8M	0.21	0.13	0.0023
Sub-WaveGrad-25	94.8M	0.21	0.11	0.0018
WaveGrad-6 [24]	15.8M	0.05	-	-
Sub-WaveGrad-6	47.4M	0.05	-	-
DiffWave-1000	1.43M	14.6	0.0025	0.0003
Sub-DiffWave-1000	14.3M	14.6	0.0021	0.0002
DiffWave(d)-50 [25]	2.62M	0.74	0.0055	0.0005
DiffWave-50	1.43M	0.73	0.0059	0.0006
Sub-DiffWave-50	12.9M	0.73	0.0051	0.0004
DiffWave-25	1.43M	0.36	0.13	0.0020
Sub-DIffWave-25	8.58M	0.36	0.12	0.0015
DiffWave-6	1.43M	0.09	-	-
Sub-DiffWave-6	4.29M	0.09	-	-

3.3. Spectral enhancement post-filtering

As described in [24], the high-frequency detail of synthesized speech cannot be reconstructed with a noise schedule that includes superfluous noise, especially when the number of iterations is relatively small. To recover the degraded high frequency component, a time-invariant spectral enhancement post-filtering is introduced. The post-filter is calculated from the averaged amplitude spectrum difference between original and synthesized waveforms using a development set. The amplitude spectra are obtained using a short-time Fourier transform (STFT) with a Hann window, and the post-filter is implemented as a simple FIR filter using the inverse STFT of the averaged amplitude spectrum difference. For example, the frequency response of the proposed post-filter as well as the spectrograms for the proposed DiffWave with six iterations are plotted in Fig. 3. In preliminary experiments, the proposed spectral enhancement post-filtering is shown to substantially improve the synthesis quality of the diffusion probabilistic vocoders.

4. EXPERIMENTS

4.1. Experimental conditions

To evaluate the proposed sub-modeling, objective and subjective experiments were conducted using a Japanese female speech corpus (neutral data) with a sampling frequency of 24 kHz. A total of 25,046 (18 h) and 20 utterances were respectively used for the training and test sets, as in [29, 30]. Additionally, 40 and 10 utterances were introduced as the development set for calculating the proposed spectral enhancement post-filters and exploring the noise schedules of six iterations, as in [24]. Sub-WaveGrad and Sub-DiffWave were compared with vanilla WaveGrad and DiffWave as well as the conventional WaveGlow [10] and PWG [17] non-autoregressive neural vocoders under both analysis–synthesis and TTS conditions.

Mel-spectrograms were used as input acoustic features. To adjust the conventional models [10, 17], 80-dimensional log-mel-spectrograms were analyzed every 12.5 ms over a Hann window with a length of 85.3 ms and a frequency band of 125–7,600 Hz, as used in [10, 17, 29, 30]. In WaveGlow, all the model parameters were the same as those used in [10, 29, 30]. PWG was trained under the same conditions in [17]. In the diffusion probabilistic vocoders, in addition to models conditioned on continuous noise levels, WaveGrad and DiffWave conditioned on discrete noise levels of 50 iterations with Linear(1×10^{-4}, 0.05, 50), commonly investigated in both vanilla WaveGrad and DiffWave, were trained for direct comparison. The WaveGrad-based models and DiffWave-based models were the same network structures as those of WaveGrad base [24] and DiffWave with 64 residual channels [25]. The batch size, batch length, and number of parameter update iterations for the diffusion probabilistic vocoders were 16, 15,900, and 1M, respectively, as for vanilla DiffWave [25]. The learning rate of Sub-WaveGrad was 0.0001 and that for the other diffusion probabilistic vocoders was 0.0002. In WaveGrad and Sub-WaveGrad, gradient clipping was introduced with a weight of 1.0.

In the inference, 1000, 50, 25, and 6 iterations were evaluated (Fig. 2). As in [24], six-iteration inference schedules were also explored by sweeping βs over $\{1, 2, 3, 4, 5, 6, 7, 8, 9\} \times 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$ using the same criterion as [24]. The results in Fig. 2 show that the numbers of sub-models for 1000, 50, 25, and 6 iterations were 10, 9, 6, and 3, respectively. The proposed spectral enhancement post-filtering was applied to all the diffusion probabilistic vocoders, where the FIR filter length, analysis window length, and shift for STFT were 512, 512, and 256 samples, respectively.

In the TTS condition, a Tacotron-based stable acoustic model with phoneme alignment, BLSTM+Taco2Dec [29, 30], was introduced with hidden Markov model-based forced alignment [31]. The feedforward Transformer-based duration predictor used in [32] was modified for full-context label input and introduced. Although

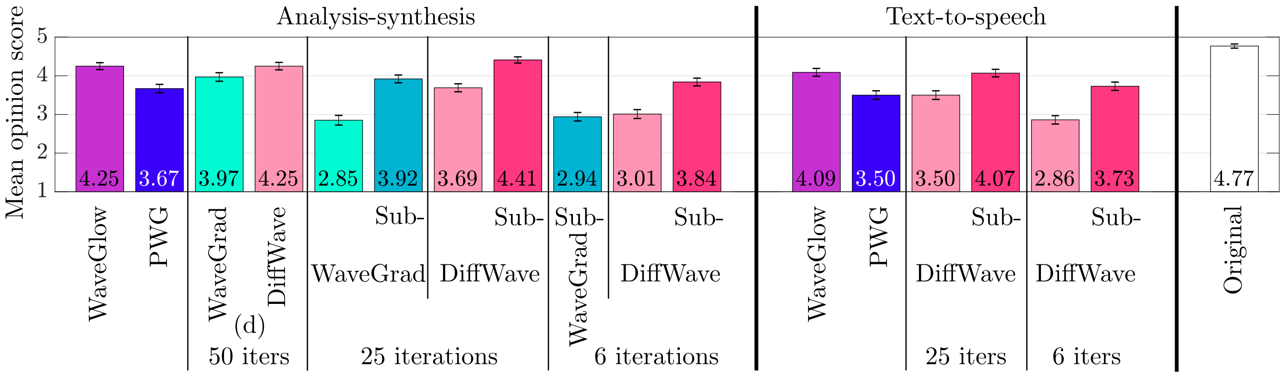


Fig. 4. Results of the MOS test with 15 listening subjects. The confidence level of the error bars is 95 %. PWG: Parallel WaveGAN, (d): discrete noise condition for WaveGrad and DiffWave. "Sub-" indicates Sub-WaveGrad and Sub-DiffWave.

the number of dimensions of the linguistic feature vectors was 130 in [29,30], it was reduced to 48 in the experiments without degrading the synthesis quality.

All the training steps and inferences were implemented using PyTorch. A RAdam optimizer [33] was introduced in all neural network models except WaveGlow. All the models were trained using four NVIDIA Tesla V100 GPUs. The training durations of the models based on WaveGlow, PWG, WaveGrad (each sub-model), and DiffWave (each sub-model) were about 20 days, 1 day, 1 day, and 2 days, respectively. Those of the WaveGrad- and DiffWave-based models using a single NVIDIA Tesla V100 GPU were about 3 days and 4 days, respectively.

As an objective evaluation, the total averaged test set loss scores of diffusion probabilistic vocoders were calculated by Eq. (1), as in training, with $n = N$ to 1. Additionally, to evaluate the model accuracy during the early iterations, the averaged test set loss scores for the first 20 % of the iterations were also measured with $n = N$ to $0.8N + 1$ for fixed numbers of iterations of 1000, 50, and 25. Furthermore, the RTFs of all models were measured using an NVIDIA Tesla V100 GPU.

To subjectively evaluate the speech waveforms synthesized by these non-autoregressive neural vocoders under analysis–synthesis and TTS conditions, mean opinion score (MOS) tests with a five-point scale [34] were conducted. These were presented through headphones to 15 Japanese adult native speakers without hearing loss (20 utterances $\times$ 18 conditions, as shown in Fig. 4, including the original test set waveforms $=$ 360 utterances).

4.2. Results and discussion

The results of the RTFs, numbers of model parameters, total averaged test set loss scores and averaged test set loss scores for the first 20 % iterations are shown in Table 1. The RTFs of the acoustic model and duration predictor were 0.015 and 0.0007, respectively. The results of the MOS tests are plotted in Fig. 4.

The results of the averaged test set loss scores show that the proposed sub-modeling improves the model accuracy in both Sub-WaveGrad and Sub-DiffWave. Furthermore, the results of the MOS tests indicate that the synthesis qualities of Sub-WaveGrad and Sub-DiffWave are substantially better than those of the vanilla models. In particular, Sub-DiffWave with 25 iterations substantially outperformed WaveGlow under the analysis–synthesis condition and was equivalent to WaveGlow under the TTS condition. Although the number of model parameters of WaveGlow is huge and 20 days are required for training using four GPUs, the proposed Sub-DiffWave with 25 iterations can be realized using six sub-models with fewer model parameters and trained for four days with six GPUs. Although Sub-DiffWave with six iterations could not reach the performance it did with 25 iterations, it still outperformed PWG under both the analysis–synthesis and TTS conditions and realized real-time synthesis with an RTF of 0.09.

Although the total averaged test set loss score of Sub-WaveGrad with 25 iterations was lower than those of the others with 25 iterations, the synthesis quality of Sub-WaveGrad with 25 iterations was lower than that of Sub-DiffWave. In contrast, the averaged test set loss scores for the first 20 % iterations of Sub-DiffWave with 25 iterations was lower than those of the others and realized higher quality synthesis. This is because noise prediction accuracy in the early iterations is important to avoid prediction errors in later iterations. These results also suggest that the GAN-TTS-based WaveGrad structure is suited for later iterations and the WaveNet-based DiffWave structure is suited for early iterations. For the same reason, the synthesis quality of DiffWave conditioned on discrete noise levels with 50 iterations was higher than that of WaveGrad. Therefore, mixture models using Sub-WaveGrad for later iterations and Sub-DiffWave for early iterations, as "DiffWaveGrad," might further improve the synthesis quality and will be explored in future work.

Consequently, the results of the objective and subjective experiments demonstrated that the proposed sub-modeling for diffusion probabilistic vocoders can substantially improve the model accuracy and synthesis quality while retaining the synthesis speed.

The optimal division criteria of sum-models, optimization of model structure for each sub-model, and mixture models of Sub-WaveGrad and Sub-DiffWave as "DiffWaveGrad" should be investigated to further improve the synthesis accuracy and inference speed.

5. CONCLUSIONS

This paper proposed noise level-limited sub-modeling for diffusion probabilistic vocoders. DiffWave conditioned on continuous noise levels and spectral enhancement post-filtering were also presented. The proposed Sub-WaveGrad and Sub-DiffWave were implemented using 10 sub-models and were separately trained with different limited noise levels. Only the necessary sub-models are then used according to the noise schedule in inference. The results of the experiments demonstrated that both the proposed Sub-WaveGrad and Sub-DiffWave models outperformed vanilla models in terms of model accuracy and synthesis quality without reducing synthesis speed.

6. REFERENCES

[1] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “WaveNet: A generative model for raw audio,” in *Proc. SSW9*, Sept. 2016, p. 125.

[2] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, RJ Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, “Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions,” in *Proc. ICASSP*, Apr. 2018, pp. 4779–4783.

[3] A. Tamamori, T. Hayashi, K. Kobayashi, K. Takeda, and T. Toda, “Speaker-dependent WaveNet vocoder,” in *Proc. Interspeech*, Aug. 2017, pp. 1118–1122.

[4] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, “Efficient neural audio synthesis,” in *Proc. ICML*, July 2018, pp. 2415–2424.

[5] J.-M. Valin and J. Skoglund, “LPCNet: Improving neural speech synthesis through linear prediction,” in *Proc. ICASSP*, May 2019, pp. 5826–7830.

[6] Q. Tian, Z. Zhang, H. Lu, L.-H. Chen, and S. Liu, “FeatherWave: An efficient high-fidelity neural vocoder with multi-band linear prediction,” in *Proc. Interspeech*, Oct. 2020, pp. 195–199.

[7] D. Rezende and S. Mohamed, “Variational inference with normalizing flows,” in *Proc. ICML*, July 2015, pp. 1530–1538.

[8] A. van den Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. van den Driessche, E. Lockhart, L. C. Cobo, F. Stimberg, N. Casagrande, D. Grewe, S. Noury, S. Dieleman, E. Elsen, N. Kalchbrenner, H. Zen, A. Graves, H. King, T. Walters, D. Belov, and D. Hassabis, “Parallel WaveNet: Fast high-fidelity speech synthesis,” in *Proc. ICML*, July 2018, pp. 3915–3923.

[9] W. Ping, K. Peng, and J. Chen, “ClariNet: Parallel wave generation in end-to-end text-to-speech,” in *Proc. ICLR*, May 2019.

[10] R. Prenger, R. Valle, and B. Catanzaro, “WaveGlow: A flow-based generative network for speech synthesis,” in *Proc. ICASSP*, May 2019, pp. 3617–3621.

[11] S. Kim, S.-G. Lee, J. Song, J. Kim, and S. Yoon, “FloWaveNet : A generative flow for raw audio,” in *Proc. ICML*, June 2019, pp. 3370–3378.

[12] K. Peng, W. Ping, Z. Song, and K. Zhao, “Non-autoregressive neural text-to-speech,” in *Proc. ICML*, July 2020.

[13] W. Ping, K. Peng, K. Zhao, and Z. Song, “WaveFlow: A compact flow-based model for raw audio,” in *Proc. ICML*, July 2020.

[14] H.-y. Lee P.-c. Hsu, “WG-WaveNet: Real-time high-fidelity speech synthesis without GPU,” in *Proc. Interspeech*, Oct. 2020, pp. 210–214.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Proc. NIPS*, Dec. 2014, pp. 2672–2680.

[16] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. C Courville, “MelGAN: Generative adversarial networks for conditional waveform synthesis,” in *Proc. NeurIPS*, Dec. 2019, pp. 14910–14921.

[17] R. Yamamoto, E. Song, and J.-M. Kim, “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, May 2020, pp. 6199–6203.

[18] M. Bińkowski, J. Donahue, S. Dieleman, A. Clark, E. Elsen, N. Casagrande, L. C. Cobo, and K. Simonyan, “High fidelity speech synthesis with adversarial networks,” in *Proc. ICLR*, Apr. 2020.

[19] J. Donahue, S. Dieleman, M. Bińkowski, E. Elsen, and K. Simonyan, “End-to-end adversarial text-to-speech,” in *Proc. ICLR*, May 2021.

[20] J. Yang, J. Lee, Y. Kim, H. Cho, and I. Kim, “VocGAN: A high-fidelity real-time vocoder with a hierarchically-nested adversarial network,” in *Proc. Interspeech*, Oct. 2020, pp. 200–204.

[21] J. Kong, J. Kim, and J. Bae, “HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis,” in *Proc. NeurIPS*, Dec. 2020.

[22] G. Yang, S. Yang, K. Liu, P. Fang, W. Chen, and L. Xie, “Multi-band MelGAN: Faster waveform generation for high-quality text-to-speech,” in *Proc. SLT*, Jan. 2021.

[23] X. Wang, S. Takaki, and J. Yamagishi, “Neural source-filter waveform models for statistical parametric speech synthesis,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 402–415, 2020.

[24] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, “WaveGrad: Estimating gradients for waveform generation,” in *Proc. ICLR*, May 2021.

[25] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, “DiffWave: A versatile diffusion model for audio synthesis,” in *Proc. ICLR*, May 2021.

[26] P. Vincent, “A connection between score matching and denoising autoencoders,” *Neural Comput.*, vol. 23, no. 7, pp. 1661–1674, July 2011.

[27] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Proc. NeurIPS*, Dec. 2020.

[28] Y. Song and S. Ermon, “Generative modeling by estimating gradients of the data distribution,” in *Proc. NeurIPS*, Dec. 2019, pp. 11918–11930.

[29] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, “Tacotron-based acoustic model using phoneme alignment for practical neural text-to-speech systems,” in *Proc. ASRU*, Dec. 2019, pp. 214–221.

[30] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, “Tranformer-based text-to-speech with weighted forced attention,” in *Proc. ICASSP*, May 2020, pp. 6729–6733.

[31] D. T. Toledano, L. A. H. Gómez, and L. V. Grande, “Automatic phonetic segmentation,” *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 617–625, Nov. 2003.

[32] Z. Zeng, J. Wang, N. Cheng, T. Xia, and Jing Xiao, “AlignTTS: Efficient feed-forward text-to-speech system without explicit alignment,” in *Proc. ICASSP*, May 2020, pp. 6741–6718.

[33] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, “On the variance of the adaptive learning rate and beyond,” in *Proc. ICLR*, Apr. 2020.

[34] ITU-T Recommendation P. 800, *Methods for subjective determination of transmission quality*, 1996.