Available online at www.sciencedirect.com

ScienceDirect

Computer Speech & Language 58 (2019) 347–363

www.elsevier.com/locate/csl

Vocoder-free text-to-speech synthesis incorporating generative adversarial networks using low-/multi-frequency STFT amplitude spectra

Yuki Saito, Shinnosuke Takamichi, Hiroshi Saruwatari

Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113−8656, Japan

Received 10 November 2018; received in revised form 28 March 2019; accepted 24 May 2019
Available online 1 June 2019

Abstract

This paper proposes novel training algorithms for vocoder-free text-to-speech (TTS) synthesis based on generative adversarial networks (GANs) that compensate for short-term Fourier transform (STFT) amplitude spectra in low/multi frequency resolution. Vocoder-free TTS using STFT amplitude spectra can avoid degradation of synthetic speech quality caused by the vocoder-based parameterization used in conventional TTS. Our previous work for the vocoder-based TTS proposed a method for incorporating the GAN-based distribution compensation into acoustic model training to improve synthetic speech quality. This paper extends the algorithm to the vocoder-free TTS and propose a GAN-based training algorithm using low-frequency-resolution amplitude spectra to overcome the difficulty in modeling complicated distribution of the high-dimensional spectra. In the proposed algorithm, amplitude spectra are transformed into low-frequency-resolution amplitude spectra by applying an average pooling function along with a frequency axis; then the GAN-based distribution compensation is performed in the low-frequency-resolution domain. Because the low-frequency-resolution amplitude spectra approximately emulate filter banks, the proposed algorithm is expected to improve synthetic speech quality by reducing differences in spectral envelopes of natural and synthetic speech. Furthermore, various frequency scales that are related to human speech perception (e.g., mel and inverse mel frequency scales) can be introduced to the proposed training algorithm by applying an frequency warping function to amplitude spectra. This paper also proposes a GAN-based training algorithm using multi-frequency-resolution amplitude spectra that uses both low- and original-frequency-resolution amplitude spectra to reduce the differences in not only spectral envelopes but also fine structures. Experimental results demonstrate that (1) GANs using low-frequency-resolution amplitude spectra improve speech quality and work robustly against the settings of the frequency resolution and hyperparameters, (2) in comparison among low-, original-, and multi-frequency-resolution amplitude spectra, the use of low-frequency-resolution ones work best improve the synthetic speech quality, and (3) the use of the inverse mel frequency scale for obtaining low-frequency-resolution amplitude spectra further improves synthetic speech quality.

Keywords: Vocoder-free text-to-speech; Training algorithm; STFT amplitude spectra; Generative adversarial networks; Frequency resolution; Frequency warping

E-mail address: yuki_saito@ipc.i.u-tokyo.ac.jp (Y. Saito).

https://doi.org/10.1016/j.csl.2019.05.008
0885-2308/

1. Introduction

Text-to-speech (TTS) synthesis (Sagisaka, 1988) is a technique for synthesizing human speech from linguistic information artificially. TTS based on statistical parametric speech synthesis (SPSS) (Zen et al., 2009) with vocoder systems has been widely investigated because it can easily control the characteristics of synthetic speech. In the conventional TTS, acoustic models learn the relationship between linguistic features and vocoder parameters that represent characteristics of the vocal cord (i.e., excitation parameters) and vocal tract (i.e., spectral parameters). In the training stage, several criteria such as the mean squared error (MSE) (Zen et al., 2013) and minimum generation error (MGE) (Wu and King, 2016) are used. In the synthesis stage, vocoder parameters are generated from the trained acoustic models and a speech waveform is synthesized by using high-quality vocoder systems such as STRAIGHT (Kawahara et al., 1999) and WORLD (Morise et al., 2016). These vocoder systems have an important role in the conventional TTS (especially, hidden Markov model-based ones (Zen and Toda, 2005; Tokuda et al., 2013) and an early stage of deep neural network (DNN)-based ones Zen et al., 2013). However, quality degradation of synthetic speech caused by the vocoder-based parameterization in state-of-the-art DNN-based TTS has become a critical problem.

Vocoder-free TTS offers a way to avoid the quality degradation by using low-level features before the vocoder-based parameterization such as short-term Fourier transform (STFT) amplitude spectra (Takaki et al., 2017) and speech waveforms (Oord et al., 2016; 2017; Mehri et al., 2017; Sotelo et al., 2017). This paper focuses on the vocoder-free TTS using STFT amplitude spectra because it can incorporate DNN-based signal-processing techniques applied in the time-frequency domain such as speech enhancement that captures the acoustic context information along the two domains for training DNNs (Xu et al., 2015) and speech separation that estimates time-frequency masking for beamforming (Wang and Chen, 2018) into the acoustic model training. In the vocoder-free TTS, STFT amplitude spectra are generated by acoustic models, and a speech waveform is synthesized from the amplitude spectra and phase spectra reconstructed by using the Griffin and Lim's method (Griffin and Lim, 1984). This framework can avoid synthesizing buzzy speech caused by the conventional vocoding process. However, the generated amplitude spectra tend to be over-smoothed (Takaki et al., 2017) as well as the generated vocoder parameters used in the conventional TTS (Toda et al., 2016), quality of the synthetic speech is significantly degraded. To address the over-smoothing effect, we previously proposed a training algorithm (Saito et al., 2017; 2018a) based on generative adversarial networks (GANs) (Goodfellow et al., 2014). The algorithm considers the adversarial loss to reduce the difference between the distributions of natural and generated speech parameters, which can effectively alleviate the over-smoothing effect and can significantly improve synthetic speech quality without any post-processing methods such as global variance compensation (Toda et al., 2007), modulation spectrum compensation (Takamichi et al., 2016; Ling et al., 2016), and GAN-based post-filtering (Kaneko et al., 2017a). Although the algorithm can be directly applied to the vocoder-free TTS, we must deal with the difficulty caused by the complicated distribution of high-dimensional amplitude spectra that represent both the vocoder-derived spectral and excitation parameters.

This paper proposes a novel algorithm based on GANs using *low-frequency-resolution amplitude spectra* to train acoustic models of the vocoder-free TTS. By using an average-pooling function along with a frequency axis, amplitude spectra are transformed into low-frequency-resolution amplitude spectra. The proposed training criterion for the acoustic models is the weighted sum of the MSE between natural and generated amplitude spectra in the original frequency resolution and the adversarial loss calculated by using the generated amplitude spectra in the low frequency resolution. Using the GANs in the low frequency resolution is similar to reducing the difference between spectral envelopes of natural and synthetic speech because the low-frequency-resolution spectra approximately emulate filter banks. Since spectral envelopes are dominant features in perceiving the quality of synthetic speech and our previous study (Saito et al., 2018a) indicated that the GAN-based distribution compensation is particularly effective for spectral parameters (i.e., mel-cepstral coefficients), the proposed algorithm is expected to improve synthetic speech quality better than using the GANs in the original frequency resolution. Furthermore, various frequency scales that are related to human speech perception (e.g., mel and inverse mel frequency scales) can be introduced to the algorithm by applying an frequency warping function to amplitude spectra. This paper also proposes a training algorithm based on GANs using *multi-frequency-resolution amplitude spectra* that uses both low- and original-frequency-resolution amplitude spectra to compensate for not only the differences in *rough* structures (i.e., spectral envelopes) but also in *fine* structures of natural and generated amplitude spectra. Experimental results demonstrate that (1) GANs using low-frequency-resolution amplitude spectra improve synthetic speech quality and work robustly against the settings of the frequency resolution and hyperparameters to control the weight for the adversarial loss (as also shown in Saito et al., 2018b), (2) in comparison among low-, original-, and multi-frequency-resolution amplitude spectra, the use of low-frequency-resolution ones are the best for improving synthetic speech quality (as also shown in Saito et al., 2018b), and (3) GANs using low-frequency-resolution amplitude spectra with an inverse mel frequency scale outperform the others with linear and mel frequency scales.

This paper is organized as follows. Section 2 briefly reviews conventional TTS frameworks: vocoder-free TTS using STFT spectra and our GAN-based training algorithm using vocoder parameters. Section 3 proposes the GAN-based training algorithms for the vocoder-free TTS. Section 4 presents experimental evaluations. Section 5 concludes this paper. Note that this paper is partially based on an international conference paper (Saito et al., 2018b) written by the authors. The contribution of this paper is that we propose a new method for using different frequency scales that are related to human speech perception in the GAN-based training algorithm and report effectiveness of the method. We also conduct additional experiments to analyze and clarify the essentials of the proposed algorithms.

2. Conventional TTS frameworks

2.1. *Vocoder-free TTS using STFT spectra*

In the vocoder-free TTS using STFT spectra (Takaki et al., 2017), DNN-based acoustic models generate STFT amplitude spectra from given linguistic features. The DNNs are trained to minimize a loss function calculated by using natural and generated amplitude spectra. Let $\vec{y}$ be a natural amplitude spectra sequence $[\vec{y}_1^\top, \ldots, \vec{y}_t^\top, \ldots, \vec{y}_T^\top]^\top$ and $\hat{\vec{y}}$ be a generated amplitude spectra sequence $[\hat{\vec{y}}_1^\top, \ldots, \hat{\vec{y}}_t^\top, \ldots, \hat{\vec{y}}_T^\top]^\top$, where t and T denote the frame index and total frame length, respectively. Let $\vec{y}_t = [y_t(1), \ldots, y_t(f), \ldots, y_t(F)]^\top$ denote an amplitude spectra vector at frame t, where f and F indicate the frequency index and the number of frequency bins from 0 Hz to the Nyquist frequency, respectively. In the training stage, the acoustic models are trained to minimize the loss function defined as the MSE between natural and generated amplitude spectra as follows:

$$L_{\mathrm{MSE}}\left(\vec{y}, \hat{\vec{y}}\right) = \frac{1}{T}\left(\hat{\vec{y}} - \vec{y}\right)^\top \left(\hat{\vec{y}} - \vec{y}\right). \tag{1}$$

In the synthesis stage, amplitude spectra are predicted by the trained acoustic models from given linguistic features; then phase spectra are reconstructed by using Griffin and Lim's method (Griffin and Lim, 1984). The synthetic speech waveform is synthesized by the inverse STFT using the amplitude and phase spectra.

2.2. *GAN-based training algorithm using vocoder parameters (Saito et al., 2018a)*

In our previous study (Saito et al., 2018a), discriminative models $D(\vec{\cdot})$ that distinguish natural and synthetic speech are introduced to the training of acoustic models in the same manner as GANs (Goodfellow et al., 2014). The discriminative models and acoustic models that generate the vocoder parameters are iteratively optimized during the training. First, discriminative models are trained to minimize the following loss:

$$L_{\mathrm{D}}\left(\vec{y}, \hat{\vec{y}}\right) = L_{\mathrm{D},1}(\vec{y}) + L_{\mathrm{D},0}\left(\hat{\vec{y}}\right), \tag{2}$$

$$L_{\mathrm{D},1}(\vec{y}) = -\frac{1}{T}\sum_{t=1}^{T} \log D(\vec{y}_t), \tag{3}$$

$$L_{\mathrm{D},0}\left(\hat{\vec{y}}\right) = -\frac{1}{T}\sum_{t=1}^{T} \log\left(1 - D\left(\hat{\vec{y}}_t\right)\right), \tag{4}$$

where $L_{\mathrm{D},1}(\vec{y})$ and $L_{\mathrm{D},0}\left(\hat{\vec{y}}\right)$ are the loss functions for natural and synthetic speech, respectively. The backpropagation algorithm is used to train $D(\vec{\cdot})$ to output 1 for natural speech and 0 for synthetic speech. After that, the acoustic models are trained to minimize the following loss:

$$L_{\mathrm{G}}\left(\vec{y},\vec{\hat{y}}\right) = L_{\mathrm{MSE}}\left(\vec{y},\vec{\hat{y}}\right) + \omega_{\mathrm{D}}\frac{\mathbb{E}_{\vec{\hat{y}}}[L_{\mathrm{MSE}}]}{\mathbb{E}_{\vec{\hat{y}}}[L_{\mathrm{ADV}}]}L_{\mathrm{ADV}}\left(\vec{\hat{y}}\right), \tag{5}$$

where $L_{\mathrm{ADV}}(\vec{\hat{y}}) = L_{\mathrm{D},1}(\vec{\hat{y}})$ is the adversarial loss that makes the discriminative models output 1 for synthetic speech. The algorithm reduces the difference between the distributions of natural and generated speech parameters.[1] ω_{D} is a hyperparameter for controlling the effect of the adversarial loss. $\mathbb{E}\vec{\hat{y}}[L_{\mathrm{MSE}}]$ and $\mathbb{E}\vec{\hat{y}}[L_{\mathrm{ADV}}]$ are the expectation values of $L_{\mathrm{MSE}}(\vec{\cdot})$ and $L_{\mathrm{ADV}}(\vec{\cdot})$, respectively. Their ratio normalizes the scale of the two loss functions.

3. Proposed GAN-based training algorithms for vocoder-free TTS

3.1. Training algorithm based on GANs using low-frequency-resolution amplitude spectra

The GAN-based training algorithm described in Section 2.2 can be directly applied to the vocoder-free TTS using STFT amplitude spectra. However, in training acoustic models, we must deal with the difficulty due to the high dimensionality of the amplitude spectra. For example, if we set the FFT length to 1,024 samples, 513-dimensional amplitude spectra are obtained, whose dimensionality is much higher than the conventional vocoder parameters. Moreover, since amplitude spectra represent both the spectral parameters and excitation parameters that are separated in the conventional TTS using vocoder systems, their distributions become more complicated than those of the vocoder parameters. To overcome these difficulties, we introduce low-frequency-resolution discriminative models $D^{(\mathrm{L})}(\vec{\cdot})$, which distinguish natural and generated amplitude spectra in the low (i.e., rough) frequency resolution.

Let $\vec{P} = [\vec{O}_{p,F}^{\top}\ \vec{I}_{F}^{\top}\ \vec{O}_{p,F}^{\top}]^{\top}$ be a $(F+2p)T$-by-FT zero-padding matrix, where p, $\vec{O}_{p,F}$, and $\vec{I}_F$ denote the size of zero-padding, p-by-F zero matrix, and F-by-F identity matrix, respectively. A $(F+2p)$-dimensional zero-padded amplitude spectra vector at frame t, i.e., $[\vec{0}_p^{\top}, \vec{y}_t^{\top}, \vec{0}_p^{\top}]^{\top}$, is calculated as $\vec{P}\vec{y}_t$, where $\vec{0}_p$ denotes a p-dimensional zero vector. Let $\vec{W}$ be a $F^{(\mathrm{L})}$-by-$(F+2p)$ pooling matrix. $F^{(\mathrm{L})}$ is the total number of frequency bins calculated as

$$F^{(\mathrm{L})} = \frac{F+2p-w}{s}+1, \tag{6}$$

where w and s denote the width and stride of the pooling operation, respectively. A low-frequency-resolution amplitude spectra vector at frame t, i.e., $\vec{y}_t^{(\mathrm{L})} = [y_t^{(\mathrm{L})}(1), \ldots, y_t^{(\mathrm{L})}(F^{(\mathrm{L})})]^{\top}$, is calculated as $\vec{W}\vec{P}\vec{y}_t$. By using the matrices $\vec{P}$ and $\vec{W}$, we define an average-pooling function $\vec{\phi}(\vec{\cdot})$ that transforms amplitude spectra in the original frequency resolution $\vec{y}$ into those in the low frequency resolution $\vec{y}^{(\mathrm{L})}$ as $\vec{\phi}(\vec{y}) = [(\vec{W}\vec{P}\vec{y}_1)^{\top}, \ldots, (\vec{W}\vec{P}\vec{y}_T)^{\top}]^{\top}$. Fig. 1 shows the matrix computation used in the average-pooling function. The above processes can be regarded as conversion from raw amplitude spectra into filter-bank parameters that represent spectral envelopes of speech.

The proposed loss function for training acoustic models is defined as follows:

$$L_{\mathrm{G}}^{(\mathrm{Low})}\left(\vec{y},\vec{\hat{y}}\right) = L_{\mathrm{MSE}}\left(\vec{y},\vec{\hat{y}}\right) + \omega_{\mathrm{D}}^{(\mathrm{L})}\frac{\mathbb{E}_{\vec{\hat{y}}}[L_{\mathrm{MSE}}]}{\mathbb{E}_{\vec{\hat{y}}^{(\mathrm{L})}}[L_{\mathrm{ADV}}]}L_{\mathrm{ADV}}\left(\vec{\hat{y}}^{(\mathrm{L})}\right), \tag{7}$$

where $\vec{\hat{y}}^{(\mathrm{L})} = \vec{\phi}(\vec{\hat{y}})$ denotes generated spectra in the low frequency resolution and $\omega_{\mathrm{D}}^{(\mathrm{L})}$ is a hyperparameter to control the effect of the second term. This loss function is formulated as the weighted sum of the MSE in the original frequency resolution and adversarial loss in the low frequency resolution. Since the distributions of amplitudes spectra in the low frequency resolution become simpler than those in the original frequency resolution, we can overcome the difficulties in modeling complicated distribution of high-dimensional amplitude spectra. Furthermore, we can expect the proposed algorithm to improve synthetic speech quality by reducing the difference between spectral envelopes of natural and synthetic speech, which are dominant features regarding the speech quality. The proposed

[1] More specifically, the difference is defined as the approximated Jensen−Shannon divergence between the two distributions.

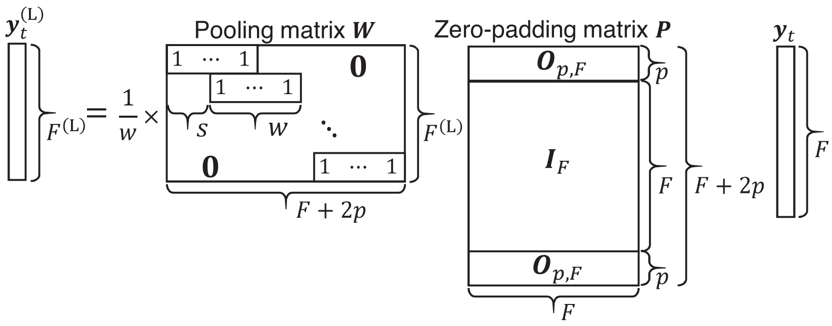

Fig. 1. Matrix computation used in average-pooling function.

low-frequency-resolution discriminative models $D^{(L)}(\vec{\cdot})$ are trained in the same manner as in minimizing Eq. (2), but $\vec{y}$ and $\hat{\vec{y}}$ in the equation are replaced with $\vec{y}^{(L)}$ and $\hat{\vec{y}}^{(L)}$, respectively.

3.2. Frequency scales for the proposed algorithm

Besides the above-mentioned approximated filter bank extraction, we can use different frequency scales that are related to human speech perception and anti-spoofing (i.e., techniques for voice spoofing detection) (Wu et al., 2016; Chen et al., 2015). For example, instead of using a linear frequency scale, we can use the mel frequency scale and its inverted version (Sahidullah et al., 2015) for the proposed training algorithms. This can be done by applying a frequency warping function to amplitude spectra before feeding them into the low-frequency-resolution discriminative models. Fig. 2 shows examples of the amplitude spectra of natural and synthetic speech in various frequency scales

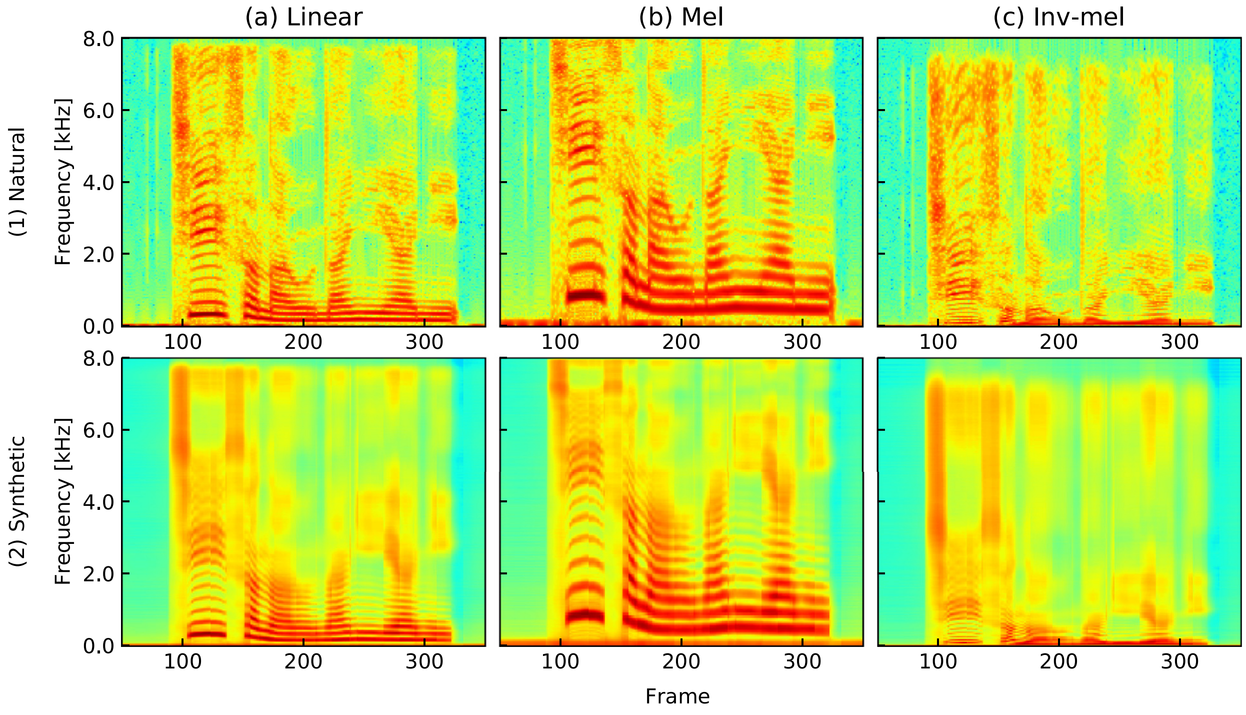

Fig. 2. Examples of amplitude spectra of natural and synthetic speech after frequency warping for (a) linear, (b) mel, and (c) inverse mel frequency scales. These spectra were extracted from one utterance of evaluation data. The synthetic amplitude spectra were generated from acoustic models trained to minimize Eq. (1).

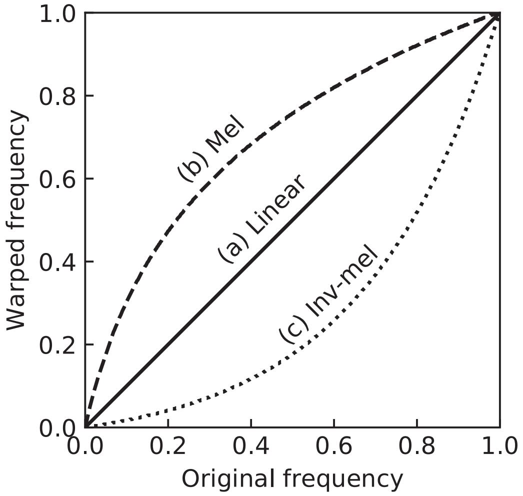

Fig. 3. Frequency warping functions for (a) linear, (b) mel, and (c) inverse mel frequency scales.

after applying the frequency warping functions shown in Fig. 3. In the mel frequency scale (Fig. 2(b)), there are fewer differences between natural and synthetic amplitude spectra compared with those in the inverse mel frequency scale, which suggests that the GAN-based distribution compensation might work well in the inverse mel frequency scale.

3.3. Training algorithm based on GANs using multi-frequency-resolution amplitude spectra

This paper also proposes a training algorithm based on GANs using multi-frequency-resolution amplitude spectra that introduce not only the low-frequency-resolution discriminative models $D^{(\mathrm{L})}(\vec{\cdot})$ but also the original-frequency-resolution discriminative models $D(\vec{\cdot})$. The proposed loss function for training acoustic models is defined as follows:

$$\begin{aligned} L_{\mathrm{G}}^{(\mathrm{Multi})}\left(\vec{y}, \hat{\vec{y}}\right) = L_{\mathrm{MSE}}\left(\vec{y}, \hat{\vec{y}}\right) + \omega_{\mathrm{D}} \frac{\mathbb{E}_{\hat{\vec{y}}}[L_{\mathrm{MSE}}]}{\mathbb{E}_{\hat{\vec{y}}}[L_{\mathrm{ADV}}]} L_{\mathrm{ADV}}\left(\hat{\vec{y}}\right) \\ + \omega_{\mathrm{D}}^{(\mathrm{L})} \frac{\mathbb{E}_{\hat{\vec{y}}}[L_{\mathrm{MSE}}]}{\mathbb{E}_{\hat{\vec{y}}^{(\mathrm{L})}}[L_{\mathrm{ADV}}]} L_{\mathrm{ADV}}\left(\hat{\vec{y}}^{(\mathrm{L})}\right). \end{aligned} \tag{8}$$

When $\omega_{\mathrm{D}} = 0$, this loss function is the same as that in Eq. (7). This algorithm can be expected to compensate for not only the differences in *rough* structures (i.e., spectral envelopes) but also in *fine* structures of natural and generated amplitude spectra. Fig. 4 illustrates the computation procedure of the loss function. Note that the two discriminative models $D^{(\mathrm{L})}(\vec{\cdot})$ and $D(\vec{\cdot})$ are separately trained.

3.4. Discussion

As described in Section 3.1, the average-pooling function used in the proposed algorithms can be regarded as the extraction of the filter-bank parameters. When we set the pooling width of w to a larger value, fine structures of the amplitude spectra get smoother. Fig. 5 shows examples of the low-frequency-resolution spectra of natural and synthetic speech with various settings of the pooling width. We can see that spectral peaks (i.e., formants) of the synthetic amplitude spectra tend to be weaker than those of the natural ones, which might be one of the causes of the speech quality degradation.

Kaneko et al. (2017b) proposed a GAN-based post-filter for STFT amplitude spectra. This post-filter-based approach requires additional computation in the synthesis stage, but our algorithms do not. Besides, as they split amplitude spectra into several sub-frequency bands and applies GANs to each band *independently*, their post-filter ignores

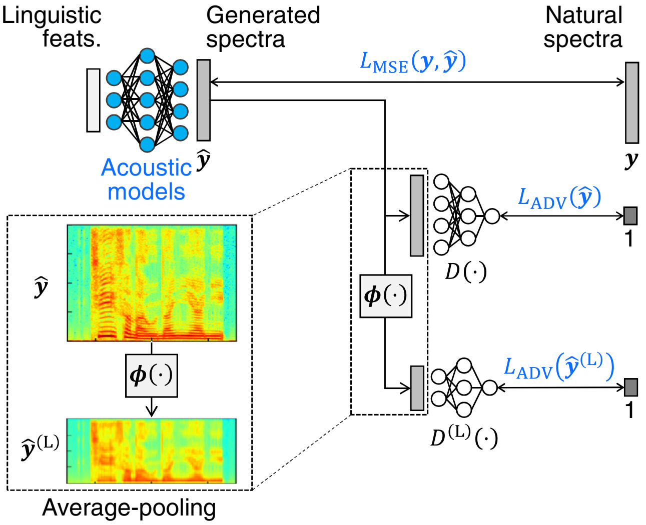
Linguistic feats.
Generated spectra
Natural spectra
$L_{\mathrm{MSE}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$
Acoustic models
$\hat{\boldsymbol{y}}$
$\boldsymbol{y}$
$L_{\mathrm{ADV}}(\hat{\boldsymbol{y}})$
1
$D(\cdot)$
$\boldsymbol{\phi}(\cdot)$
$L_{\mathrm{ADV}}(\hat{\boldsymbol{y}}^{(\mathrm{L})})$
1
$D^{(\mathrm{L})}(\cdot)$
$\hat{\boldsymbol{y}}^{(\mathrm{L})}$
Average-pooling

Fig. 4. Loss functions for updating acoustic models in proposed algorithm based on GANs using multi-frequency-resolution amplitude spectra. $\vec{\phi}(\cdot)$ is the average-pooling function for converting amplitude spectra into low-frequency-resolution spectra.

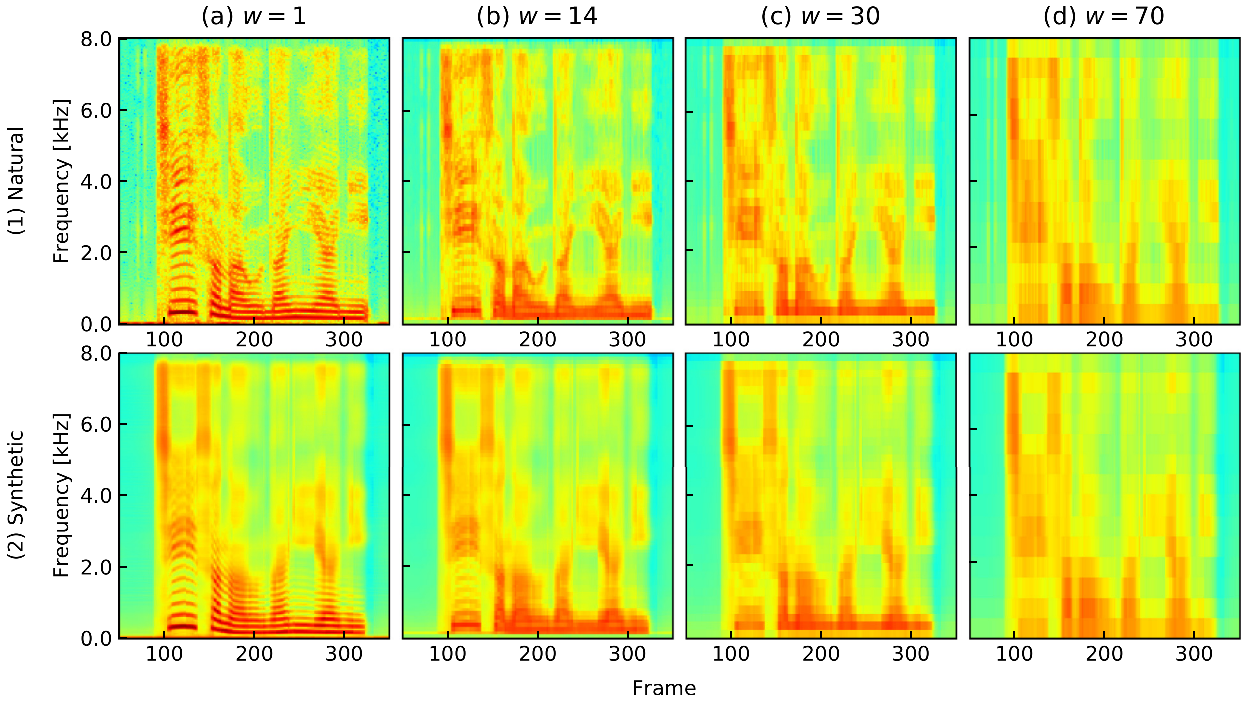

Fig. 5. Examples of low-frequency-resolution spectra of natural and synthetic speech. These spectra were extracted from one utterance of evaluation data. The size of zero-padding p and stride of pooling s were set to 6 and $w/2$, respectively. The column "(a) $w = 1$" corresponds to amplitude spectra in the original frequency resolution. The synthetic amplitude spectra were generated from acoustic models trained to minimize Eq. (1).

the overall spectral structures (i.e., spectral envelopes) and their correlation. On the other hand, our algorithms can effectively capture them by reducing the dimensionality of the spectra while preserving the whole spectral structure.

We extend our GAN-based algorithm for TTS using *vocoder parameters* (Saito et al., 2017) to the ones using *STFT amplitude spectra*. Although Juvela et al. (2018) proposed a GAN-based method for synthesizing a speech waveform from natural mel frequency cepstral coefficients, their method was not conditioned by linguistic information and cannot be directly applied to TTS. We expect the proposed algorithms to extend a GAN-based algorithm to the ones that directly synthesizes a speech waveform from linguistic features. We hope that the idea that uses GANs in the low frequency resolution can also be applied to WaveGAN and SpecGAN (McAuley and Puckette, 2018), which synthesize an audio waveform by using unconditional GANs.

4. Experimental evaluation

4.1. Experimental conditions

A speech dataset of a Japanese female speaker who uttered 4,007 sentences was used. The numbers of utterances used for training and evaluation were 3808 and 199, respectively. The sampling rate of the speech signals was 16 kHz. The frame length, shift length, and FFT length were set to 400, 80, and 1,024 samples, respectively. The Hamming window was used for STFT analysis. In the training phase, linguistic features that had a real value and log-amplitude spectra were normalized to have zero-mean unit-variance. Ninety percent of the silence frames were removed from the training data for improving training accuracy.

DNN architectures for acoustic and discriminative models were Feed-Forward networks. The inputs of the acoustic models were 444-dimensional vectors including 439-dimensional linguistic features, 3-dimensional duration features, continuous log F_0, and U/V. The F_0 was extracted from speech data by using the STRAIGHT vocoder systems (Kawahara et al., 1999). DNNs for predicting the duration and F_0 from linguistic features were constructed in advance. The architecture for the acoustic models included 3 × 1024-unit hidden layers with the rectified linear unit (ReLU) (Glorot et al., 2011) activation function and 513-unit output layer with the linear activation function. The architecture for the discriminative models in the original frequency resolution included 3 × 512-unit hidden layers with the ReLU activation function and one unit output layer with the sigmoid activation function. The architecture for the discriminative models in the low frequency resolution was almost the same as that in the original frequency resolution; i.e., the activation functions used in the hidden and output layers were ReLU and sigmoid, the number of hidden layers was 3, but the number of inputs and hidden units varied in accordance with the parameters of the pooling function $\vec{\phi}(\vec{\cdot})$. In the following experiments, we fixed $p = 6$ and $s = w/2$ in Eq. (6). The width of the pooling window w was set to 14, 30, and 70. Accordingly, the number of input units $F^{(\mathrm{L})}$ was set to 74, 34, and 14. We changed the number of the hidden units in $D^{(\mathrm{L})}(\vec{\cdot})$ to 128, 64, and 32 as the number of input units decreased.

In the training stage, the acoustic models were initialized by minimizing the MSE between natural and generated amplitude spectra described in Section 2.1. The number of iterations for initializing the acoustic models was 25. Here, “Iteration” means using all training data (3,808 utterances) once for training. The discriminative models in the original and low frequency resolution were initialized using natural amplitude spectra and ones generated by the initialized acoustic models. The number of iterations for initializing the discriminative models was 5. The proposed GAN-based training algorithms were performed with 25 iterations. The expectation values for scaling the loss functions were estimated at each iteration step. AdaGrad (Duchi et al., 2011) was used as the optimization algorithm, setting the learning rate to 0.01.

4.2. Objective evaluations

We calculated the root mean square error (RMSE) between natural and synthetic amplitude spectra and the spoofing rate (Saito et al., 2018a) of the two discriminative models $D(\vec{\cdot})$ and $D^{(\mathrm{L})}(\vec{\cdot})$ for evaluating our algorithms objectively. The spoofing rate is the number of spoofing synthetic spectra divided by the total number of synthetic spectra in the evaluation data. Here, “spoofing synthetic spectra” indicates spectra for which the discriminative models recognized them as natural ones. The discriminative models for calculating the spoofing rates were constructed using natural and generated amplitude spectra of the conventional training algorithm that minimized the MSE loss shown in Eq. (1) (Takaki et al., 2017). We compared our algorithms with the combination of the hyper parameters $(\omega_{\mathrm{D}}, \omega_{\mathrm{D}}^{(\mathrm{L})})$ setting each parameter to 0.0 or 1.0.

Table 1 shows the experimental results. From the results, we found that the algorithm with the setting $(\omega_{\mathrm{D}} = 0.0, \omega_{\mathrm{D}}^{(\mathrm{L})} = 0.0)$, i.e., the same as the conventional algorithm, achieved the lowest RMSE among the four algorithms. However, its spoofing rate of $D(\vec{\cdot})$ and $D^{(\mathrm{L})}(\vec{\cdot})$ was the lowest among the four algorithms, which suggested that it did not train the acoustic models that could fool the two discriminative models. On the other hand, our algorithm setting to $(\omega_{\mathrm{D}} = 0.0, \omega_{\mathrm{D}}^{(\mathrm{L})} = 1.0)$ generated amplitude spectra that could deceive the low-frequency-resolution discriminative models $D^{(\mathrm{L})}(\vec{\cdot})$, although the RMSE became slightly worse than that of the conventional

Table 1
Results of objective evaluations with their standard deviations for our algorithms using various hyper parameter settings $(\omega_{\mathrm{D}}, \omega_{\mathrm{D}}^{(\mathrm{L})})$. Pooling parameters used in average-pooling were set to $w = 30$, $s = 15$, and $p = 6$. Values in this table were calculated over all evaluation data.

$(\omega_{\mathrm{D}}, \omega_{\mathrm{D}}^{(\mathrm{L})})$	(0.0, 0.0)	(0.0, 1.0)	(1.0, 0.0)	(1.0, 1.0)
RMSE of amplitude spectra	1.0948 ± 0.0880	1.1247 ± 0.0916	1.2480 ± 0.0847	1.2393 ± 0.0834
Spoofing rate of $D(\vec{\cdot})$	0.0019 ± 0.0042	0.5507 ± 0.0696	0.9999 ± 0.0001	0.9999 ± 0.0005
Spoofing rate of $D^{(\mathrm{L})}(\vec{\cdot})$	0.0273 ± 0.0177	0.9704 ± 0.0269	0.9965 ± 0.0055	0.9955 ± 0.0065

algorithm. Note that the increase of the RMSE was also reported in our previous work (Saito et al., 2018a). We also found that some of the generated spectra of the proposed algorithm could fool the original-frequency-resolution discriminative models $D(\vec{\cdot})$ nevertheless deceiving the models was not considered during the acoustic model training. These results indicated that the proposed algorithm using GANs based on low-frequency-resolution amplitude spectra could reduce the differences between natural and generated amplitude spectra observed in their rough structures. Meanwhile, focusing on the proposed algorithm that used GANs with original-frequency-resolution amplitude spectra, i.e., “(1.0, 0.0)” and “(1.0, 1.0)” in Table 1, we observed that these algorithms had similar tendencies in the objective scores. Although the two algorithms improved spoofing rate of both $D(\vec{\cdot})$ and $D^{(\mathrm{L})}(\vec{\cdot})$ much higher than the other two algorithms, they also considerably degraded the RMSE of the amplitude spectra, which might deteriorate the synthetic speech quality.

4.3. *Subjective evaluations*

We conducted subjective evaluations in terms of synthetic speech quality. A series of preference AB tests was conducted to evaluate the quality of speech samples produced by using several algorithms. 25 listeners participated in each of the following evaluations by using our crowd-sourced evaluation systems, and each listener evaluated 10 samples. The total number of listeners was 1,125. In the following evaluations, “Baseline” denotes the conventional training algorithm that minimizes the MSE loss shown in Eq. (1) (Takaki et al., 2017).

4.3.1. *Evaluation of GANs using original-frequency-resolution amplitude spectra*

We investigate the effect of GAN-based training in the original frequency resolution (i.e., the same algorithm as in our previous work Saito et al. (2018a)) by fixing $\omega_{\mathrm{D}}^{(\mathrm{L})} = 0$ and by setting $\omega_{\mathrm{D}} = 0.5$ or 1.0. We compared the synthetic speech quality of “Baseline” and the algorithm using the GANs with the settings “$\omega_{\mathrm{D}} = 0.5$,” and “$\omega_{\mathrm{D}} = 1.0$.” Table 2 shows the experimental results. Compared with “Baseline,” the GAN-based algorithm significantly degraded synthetic speech quality regardless of the hyperparameter settings. Therefore, we confirmed that just using the GAN-based training algorithm, which was effective in conventional TTS with vocoder systems (Saito et al., 2018a), did not improve speech quality in the vocoder-free TTS using STFT amplitude spectra.

4.3.2. *Evaluation of GANs using low-frequency-resolution amplitude spectra*

We investigate the effect of the width of pooling window w by fixing $\omega_{\mathrm{D}} = 0$ and by setting $\omega_{\mathrm{D}}^{(\mathrm{L})} = 1$. We compared the quality of the synthetic speech of “Baseline” and the proposed algorithm using GANs using low-frequency-resolution amplitude spectra with the settings “$w = 14$,” “$w = 30$,” and “$w = 70$.” We changed the

Table 2
Preference scores of synthetic speech quality with their p-values (GANs using original-frequency-resolution amplitude spectra with various hyperparameter settings of ω_{D}).

Method A	Score	p-value	Method B
$\omega_{\mathrm{D}} = 0.5$	0.300 vs. **0.700**	$< 10^{-10}$	Baseline
$\omega_{\mathrm{D}} = 1.0$	0.280 vs. **0.720**	$< 10^{-10}$	Baseline
$\omega_{\mathrm{D}} = 0.5$	0.496 vs. **0.504**	8.6×10^{-1}	$\omega_{\mathrm{D}} = 1.0$

Table 3
Preference scores of synthetic speech quality with their p-values (GANs using low-frequency-resolution amplitude spectra with various settings of w). Here, we changed the number of hidden units in accordance with the settings of w.

Results comparing “Baseline” with the GANs

Method A	Score	p-value	Method B
$w = 14$	**0.568** vs. 0.432	2.3×10^{-3}	Baseline
$w = 30$	**0.572** vs. 0.428	1.2×10^{-3}	Baseline
$w = 70$	**0.528** vs. 0.472	2.1×10^{-1}	Baseline

Results comparing the algorithms using the GANs

Method A	Score	p-value	Method B
$w = 14$	0.488 vs. **0.512**	5.9×10^{-1}	$w = 30$
$w = 30$	**0.532** vs. 0.468	1.5×10^{-1}	$w = 70$
$w = 70$	0.472 vs. **0.528**	2.1×10^{-1}	$w = 14$

number of the hidden units in the discriminative models in accordance with the pooling parameter settings. Table 3 shows the experimental results. From the results shown in Table 3(a), we can see that the proposed algorithm always achieved better scores than “Baseline,” regardless of their settings of pooling width, which demonstrated the effectiveness of the algorithm.

Further, we conducted a subjective evaluation of the proposed algorithm using discriminative models with the fixed number of the hidden units, i.e., we set the number to 128 regardless of the pooling parameter settings. Table 4 shows the experimental results. From this table, we found that the proposed algorithm improved the synthetic speech quality whether we changed the number of hidden units in the discriminative models or not, which suggested that the algorithm worked robustly against the size of the discriminative models. We set the pooling width w to 30 in the following evaluations because the results in Tables 3(b) and 4(b) show that “$w = 30$” were the best, although there were no significant differences among the preference scores, and the number of hidden units in the discriminative models to 64 for reducing the number of parameters in the models.

We also investigated the effect of the hyperparameter in the proposed algorithm. We fixed $\omega_{\mathrm{D}} = 0$ and set $\omega_{\mathrm{D}}^{(\mathrm{L})} = 0.5$ or 1.0. We compared “Baseline” and using the GAN-based algorithm with the settings “$\omega_{\mathrm{D}}^{(\mathrm{L})} = 0.5$,” and “$\omega_{\mathrm{D}}^{(\mathrm{L})} = 1.0$.” Table 5 shows the experimental results. From the results, we concluded that the proposed GAN-based algorithm using low-frequency-resolution amplitude spectra improved the synthetic speech quality regardless of its hyperparameter settings.

Table 4
Preference scores of synthetic speech quality with their p-values (GANs using low-frequency-resolution amplitude spectra with various settings of w). Here, we fixed the number of hidden units regardless of the settings of w.

Results comparing “Baseline” with the GANs

Method A	Score	p-value	Method B
$w = 14$	**0.548** vs. 0.452	3.2×10^{-2}	Baseline
$w = 30$	**0.600** vs. 0.400	$< 10^{-6}$	Baseline
$w = 70$	**0.560** vs. 0.440	7.2×10^{-3}	Baseline

Results comparing the algorithms using the GANs

Method A	Score	p-value	Method B
$w = 14$	0.472 vs. **0.528**	2.1×10^{-1}	$w = 30$
$w = 30$	**0.528** vs. 0.472	2.1×10^{-1}	$w = 70$
$w = 70$	0.476 vs. **0.524**	2.8×10^{-1}	$w = 14$

Table 5
Preference scores of synthetic speech quality with their p-values (GANs using low-frequency-resolution amplitude spectra with various hyper-parameter settings of $\omega_{\mathrm{D}}^{(\mathrm{L})}$ and the fixed pooling width $w = 30$).

Method A	Score	p-value	Method B
$\omega_{\mathrm{D}}^{(\mathrm{L})} = 0.5$	**0.544** vs. 0.456	4.9×10^{-2}	Baseline
$\omega_{\mathrm{D}}^{(\mathrm{L})} = 1.0$	**0.588** vs. 0.412	7.6×10^{-5}	Baseline
$\omega_{\mathrm{D}}^{(\mathrm{L})} = 0.5$	**0.504** vs. 0.496	8.6×10^{-1}	$\omega_{\mathrm{D}}^{(\mathrm{L})} = 1.0$

4.3.3. Comparison between pooling and smoothing of amplitude spectra

For investigating the effectiveness of the dimensionality reduction by the average-pooling function of the proposed algorithm, we compared the use of low-frequency-resolution amplitude spectra (“Pooling”) and original-frequency-resolution amplitude spectra smoothed by a simple moving average filter (“Smoothing”). The filter size was set to 25. We can regard the smoothed spectra as the spectral envelope parameters without the dimensionality reduction. Table 6 shows the experimental results. From this table, we found that “Pooling” outperformed “Smoothing,” which suggested that the dimensionality reduction was one of the essentials for the synthetic speech quality improvement.

4.3.4. Evaluation of GANs using multi-frequency-resolution amplitude spectra

We examined the effects of the GAN-based proposed algorithm using multi-frequency-resolution amplitude spectra. We generated speech samples using the following three algorithms:

$$\begin{aligned}\text{Original: } & (\omega_{\mathrm{D}}, \omega_{\mathrm{D}}^{(\mathrm{L})}) = (1.0, 0.0)\\ \text{Low: } & (\omega_{\mathrm{D}}, \omega_{\mathrm{D}}^{(\mathrm{L})}) = (0.0, 1.0)\\ \text{Multi: } & (\omega_{\mathrm{D}}, \omega_{\mathrm{D}}^{(\mathrm{L})}) = (1.0, 1.0)\end{aligned}$$

Table 7 shows the experimental results. Obviously, “Low” achieved a much higher score than the others. To investigate the reason, we plotted amplitude spectra of the synthetic speech used for the evaluations in Fig. 6. We can see that the difference in spectral peaks between natural and synthetic speech was reduced by the GAN-based training algorithms (Figs. 6(c), (d), and (e)). However, there were some temporal discontinuities in the amplitude spectra generated by the acoustic models trained with “Original” and “Multi” (Figs. 6(d) and (e)), which might significantly degrade synthetic speech quality.

Table 6
Preference scores of synthetic speech quality with their p-values (comparison between GANs using low-frequency-resolution and smoothed original-frequency-resolution amplitude spectra).

Method A	Score	p-value	Method B
Pooling	**0.58** vs. 0.42	3.3×10^{-4}	Smoothing

Table 7
Preference scores of speech quality with their p-values (GANs based on multi-frequency-resolution amplitude spectra).

Method A	Score	p-value	Method B
Low	**0.808** vs. 0.192	$< 10^{-10}$	Multi
Multi	0.492 vs. **0.508**	7.2×10^{-1}	Original
Original	0.192 vs. **0.808**	$< 10^{-10}$	Low

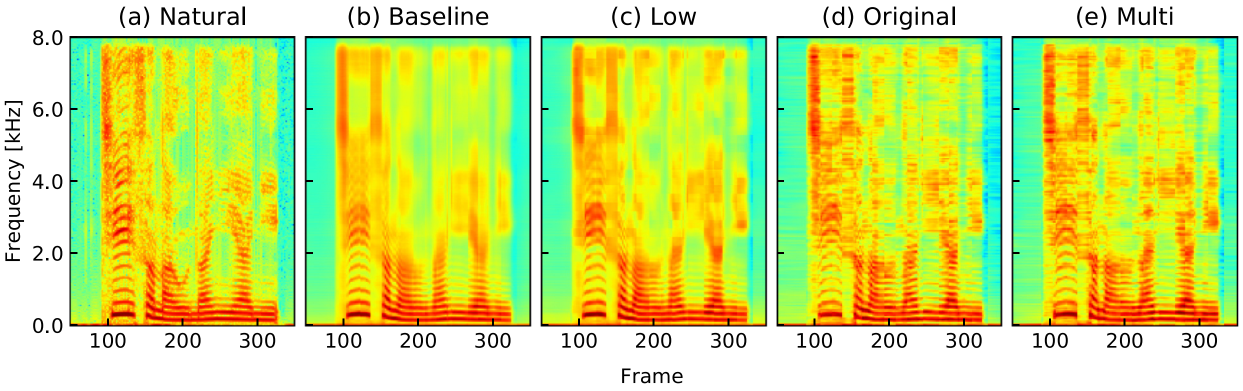

Fig. 6. Examples of amplitude spectra of (a) natural speech and synthetic speech generated by the methods named as (b) “Baseline,” (c) “Low,” (d) “Original,” and (e) “Multi.” These spectra were extracted from one utterance of evaluation data.

4.3.5. *Evaluation of DNN architectures for acoustic models used in GANs*

To deal with the temporal discontinuities observed in the generated amplitude spectra shown in Fig. 6(d), we used DNN architectures richer than Feed-Forward DNN as acoustic models. Although a straightforward way for modeling the temporal structure is to use recurrent architectures such as long-short term memory (LSTM) (Hochreiter and Schmidhuber, 1997; Zen and Sak, 2015), we used gated convolutional neural networks (CNNs) (Dauphin et al., 2017) because they are applied to sequential modeling in speech processing (Kaneko and Kameoka, 2018; Kameoka et al., 2018) and can be trained faster than the LSTM. Here, we added a 1D convolutional (Conv1D) layer along the time axis with the gated linear unit (GLU) activation function (Dauphin et al., 2017) after the output layer of the acoustic models. We also introduced a residual connection (He et al., 2016) between the output and Conv1D layers for better modeling. In the following evaluations, we set the width and zero-padding size of the convolution to 15 and 7, respectively.

First, we compared the synthetic speech quality of the four algorithms (“Baseline,” “Original,” “Low,” and “Multi”) with the gated CNNs used as the acoustic models. Table 8 shows the experimental results. From the results, we found tendencies similar to those observed in the experimental results that used Feed-Forward DNNs as acoustic models in the proposed algorithms, which indicated that the sequential modeling in the acoustic models of the proposed algorithm could not deal with the temporal discontinuities in the generated amplitude spectra.

Then, we evaluated the effectiveness of the CNN-based acoustic models in the GAN-based algorithm using low-frequency-resolution amplitude spectra. We compared two acoustic models: 1) Feed-Forward DNNs (“FFNN”) and 2) gated CNNs (“CNN”). Table 9 shows the experimental results.

From this table, we concluded that the CNN-based acoustic models were effective in improving the synthetic speech quality of the algorithm better than the Feed-Forward-DNN-based ones.

Table 8
Preference scores of synthetic speech quality with their p-values (GANs using low-frequency-resolution amplitude spectra with gated-CNN-based acoustic models).

Results comparing “Baseline” with the GANs			
Method A	Score	p-value	Method B
Original	0.180 vs. **0.820**	$< 10^{-10}$	Baseline
Low	**0.608** vs. 0.392	$< 10^{-5}$	Baseline
Multi	0.252 vs. **0.748**	$< 10^{-10}$	Baseline
Results comparing the algorithms using GANs			
Method A	Score	p-value	Method B
Low	**0.808** vs. 0.192	$< 10^{-10}$	Multi
Multi	0.496 vs. **0.504**	8.6×10^{-1}	Original
Original	0.172 vs. **0.828**	$< 10^{-10}$	Low

Table 9
Preference scores of synthetic speech quality with their p-values (comparison between Feed-Forward DNN ("FFNN") and gated CNN ("CNN") for acoustic modeling).

Method A	Score	p-value	Method B
CNN	**0.644** vs. 0.356	$< 10^{-10}$	FFNN

4.3.6. Evaluation of DNN architectures for discriminative models used in GANs

We further investigate the effect of DNN architectures for discriminative models used in the proposed algorithms. In the evaluation, we appropriated the gated CNNs used in Section 4.3.5 to the acoustic models, and compared the following four discriminative models:

FFNN-O: the same as used in "Original"
FFNN-L: the same as used in "Low"
CNN-1D: replacing average-pooling with a Conv1D layer along the frequency axis with the GLU activation function
CNN-2D: using 2D convolutional (Conv2D) layers for capturing the time-frequency structures of the spectra

We can regard the Conv1D layer in "CNN-1D" as a trainable multi-channel average-pooling function for obtaining low-frequency-resolution amplitude spectra. Table 10 shows the details of the four discriminative models.

We generated speech samples using the acoustic models based on the gated CNNs trained with the four different discriminative models. Table 11 shows the experimental results.

From the results shown in Table 11(a), we found that the CNN-based discriminative models were effective in improving the synthetic speech quality of the GAN-based algorithm using original-frequency-resolution amplitude spectra. However, as shown in Table 11(b), the CNNs could not outperform "FFNN-L." One of the reasons that caused the results might be the difficulty in the training of GANs with more complicated DNN architectures for the proposed algorithm.

4.3.7. Evaluation of frequency scale of low-frequency-resolution amplitude spectra used for GANs

We investigated the effect of the frequency scale in the GAN-based algorithm using low-frequency-resolution amplitude spectra. We used three types of frequency scales: linear, mel, and inverse mel scales. Table 12 shows the experimental results. From the results, we found that using the GANs with the mel frequency scale significantly degraded synthetic speech quality, while using them with the inverse mel frequency scale successfully improved it better than using them with the linear frequency scale. To investigate the reason, we plotted the amplitude spectra of synthetic speech used for the evaluations in Fig. 7. From the figure, the differences among the three generated amplitude spectra were observed in a higher frequency band. The generated spectra of "Low w/ inv-mel" became closer to the natural spectra compared to the others, which might be the cause of the quality improvement.

4.3.8. Evaluation of frequency band used for average-pooling

We can regard the average-pooling function in the proposed algorithms as the reduction of undesired components in amplitude spectra fed into discriminative models. Here, we split the spectra into two sub-bands, 0–2 kHz

Table 10
DNN architectures for discriminative models. Width, stride, and zero-padding size of Conv1D layer in "CNN-1D" was set to 30, 15, and 6, respectively. Width parameters of Conv2D layers in "CNN-2D" were set to 9, 7, 5, and 3. Accordingly, stride and zero-padding parameters were set to 4, 3, 2, and 1.

	Input dim.	Hidden layers	Output
FFNN-O	513	Linear 512 units × 3 (ReLU)	Frame-wise
FFNN-L	34	Linear 64 units × 3 (ReLU)	Frame-wise
CNN-1D	513	Conv1D 4 channels (GLU) + Linear 64 units × 3 (ReLU)	Frame-wise
CNN-2D	513	Conv2D 32–16–8–4 channels (ReLU) + Fully-connected	Segment-wise

Table 11
Preference scores of synthetic speech quality with their *p*-values (GANs using gated CNNs as acoustic models with various DNN architectures for discriminative models).

Results comparing GANs using original-frequency-resolution amplitude spectra			
Method A	Score	*p*-value	Method B
FFNN-O	0.376 vs. **0.624**	$< 10^{-6}$	CNN-1D
FFNN-O	0.120 vs. **0.880**	$< 10^{-10}$	CNN-2D
CNN-1D	0.348 vs. **0.652**	$< 10^{-10}$	CNN-2D
Results comparing the GAN-based algorithms using original- and low-frequency-resolution amplitude spectra			
Method A	Score	*p*-value	Method B
FFNN-L	**0.820** vs. 0.180	$< 10^{-10}$	CNN-1D
FFNN-L	**0.828** vs. 0.172	$< 10^{-10}$	CNN-2D

Table 12
Preference scores of synthetic speech quality with their *p*-values (GANs using low-frequency-resolution amplitude spectra with various frequency scale).

Results comparing "Baseline" with the GANs using mel- and inverse mel scale			
Method A	Score	*p*-value	Method B
Mel	0.392 vs. **0.608**	$< 10^{-5}$	Baseline
Inv-mel	**0.636** vs. 0.364	$< 10^{-10}$	Baseline
Results comparing the algorithms with different frequency scale			
Method A	Score	*p*-value	Method B
Linear	**0.752** vs. 0.248	$< 10^{-10}$	Mel
Mel	0.272 vs. **0.728**	$< 10^{-10}$	Inv-mel
Inv-mel	**0.576** vs. 0.424	6.5×10^{-4}	Linear

(including at least the first and second formants) and 2–8 kHz, and applied average-pooling to each sub-band individually. The results of the comparison among the combination of these two factors, frequency band (low or high) and average-pooling (with or without), should be meaningful for clarifying what component was effective. Fig. 8 illustrates a conceptual diagram of the split-and-pooling procedures.

Table 13 shows the experimental results. Here, we used Feed-Forward DNNs for both acoustic and discriminative models. Two points found from the results in this table are noteworthy: (1) using average-pooling in the *low* frequency band always achieved higher scores, and (2) there was no significant difference between average pooling in

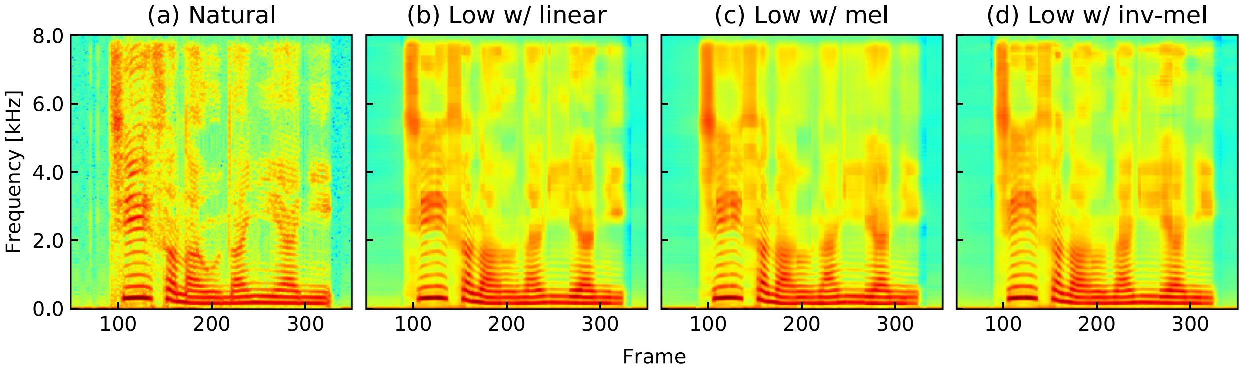

Fig. 7. Examples of amplitude spectra of (a) natural speech and synthetic speech generated by the proposed algorithms using low-frequency-resolution GANs with (b) linear scale, (c) mel scale, and (d) inverse mel scale. These spectra were extracted from one utterance of evaluation data.

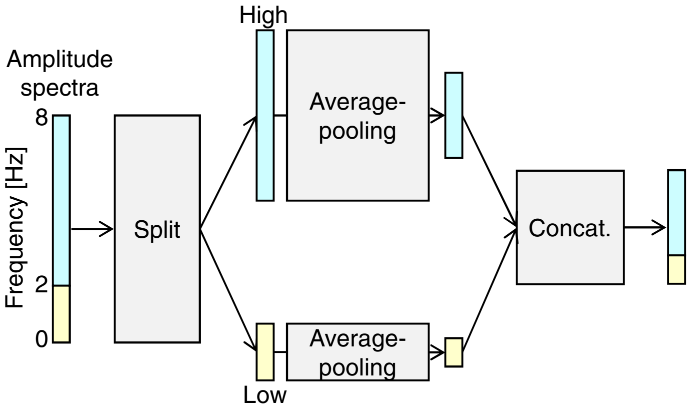

Fig. 8. Conceptual diagram of split-and-pooling procedures corresponding to the proposed algorithm with average-pooling applied to both low and high frequency bands, i.e., "(w/, w/)" in Table 13. "w/o" in Table 13 means the direct use of original-frequency-resolution amplitude spectra in the low or high frequency band.

both low and high frequency bands and that in an *only low* frequency band. These results suggested that applying the GAN-based distribution compensation to the low frequency band degraded the synthetic speech quality, which corresponded to the results shown in Table 12(a) that the use of the mel frequency scale significantly deteriorated the synthetic speech quality.

5. Conclusion

We propose two training algorithms for incorporating generative adversarial networks (GANs) into vocoder-free text-to-speech (TTS) synthesis using short-term Fourier transform (STFT) amplitude spectra. In the proposed algorithm based on GANs using low-frequency-resolution amplitude spectra, acoustic models are trained to minimize the mean squared error (MSE) between natural and generated amplitude spectra in the original frequency resolution and the distribution differences of their distributions in the low frequency resolution. This algorithm can be extended to the one based on GANs using multi-frequency-resolution amplitude spectra, which also minimizes the distribution differences of natural and generated amplitude spectra in the original frequency resolution. Experimental results indicated that the GANs using original-/multi-frequency-resolution amplitude spectra degraded synthetic speech quality, but the one using low-frequency-resolution amplitude spectra successfully improved it compared with conventional algorithm minimizing the MSE between natural and generated amplitude spectra. Moreover, we found that GANs using low-frequency-resolution amplitude spectra with the inverse mel frequency scale further improved speech quality. In the future, we will further investigate the effects of the hyperparameters of the proposed algorithms and introducing conditional GANs (Mirza and Osindero, 2014) for the algorithms.

Table 13
Preference scores of synthetic speech quality with their *p*-values (comparison among the combination of frequency band (low or high) and average-pooling (w/ or w/o)).

Method A (low, high)	Score	*p*-value	Method B (low, high)
(w/o, w/o)	0.212 vs. **0.788**	$< 10^{-10}$	(w/, w/)
(w/o, w/o)	0.160 vs. **0.840**	$< 10^{-10}$	(w/, w/o)
(w/o, w/o)	**0.668** vs. 0.332	$< 10^{-10}$	(w/o, w/)
(w/o, w/)	0.160 vs. **0.840**	$< 10^{-10}$	(w/, w/o)
(w/, w/o)	0.468 vs. **0.532**	1.5×10^{-1}	(w/, w/)
(w/, w/)	**0.844** vs. 0.156	$< 10^{-10}$	(w/o, w/)

Acknowledgments

Part of this work was supported by SECOM Science and Technology Foundation, and JSPS KAKENHI Grant Number 18J22090, and was executed under the Commissioned Research of MIC SCOPE (182103104).

References

Chen, N., Qian, Y., Dinkel, H., Chen, B., Yu, K., 2015. Robust deep feature for spoofing detection the SJTU system for ASVspoof 2015 challenge. In: Proceedings of the INTERSPEECH, Dresden, Germany, pp. 2097–2101.

Dauphin, Y.N., Fan, A., Auli, M., Grangier, D., 2017. Language modeling with gated convolutional networks. In: Proceedings of the ICML, Sydney, Australia, pp. 933–941.

Duchi, J., Hazan, E., Singer, Y., 2011. Adaptive subgradient methods for online learning and stochastic optimization. J. Mach. Learn. Res. 12, 2121–2159.

Glorot, X., Bordes, A., Bengio, Y., 2011. Deep sparse rectifier neural networks. In: Proceedings of the AISTATS, Lauderdale, USA, pp. 315–323.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. In: Proceedings of the NIPS, pp. 2672–2680.

Griffin, D., Lim, J., 1984. Signal estimation from modified short-time fourier transform. IEEE Trans. Audio, Speech, Lang. Process. 32 (2), 236–243.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the CVPR, Las Vegas, U.S.A., pp. 770–778.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Juvela, L., Bollepalli, B., Wang, X., Kameoka, H., Airaksinen, M., Yamagishi, J., Alku, P., 2018. Speech waveform synthesis from MFCC sequences with generative adversarial networks. In: Proceedings of the ICASSP, Calgary, Canada, pp. 5679–5683.

Inoue, S., Kameoka, H., Li, L., Seki, S., Makino, S., 2018. Joint separation and dereverberation of reverberant mixtures with multichannel variational autoencoder. In: Proceedings of the ICASSP, Brighton, U.K., pp. 96–100.

Kaneko, T., Kameoka, H., 2018. CycleGAN-VC: non-parallel voice conversion using cycle-consistent adversarial networks. In: Proceedings of the EUSIPCO, Rome, Italy, pp. 2114–2118.

Kaneko, T., Kameoka, H., Hojo, N., Ijima, Y., Hiramatsu, K., Kashino, K., 2017a. Generative adversarial network-based postfilter for statistical parametric speech synthesis. In: Proceedings of the ICASSP, New Orleans, USA, pp. 4910–4914.

Kaneko, T., Takaki, S., Kameoka, H., Yamagishi, J., 2017b. Generative adversarial network-based postfilter for STFT spectrograms. In: Proceedings of the INTERSPEECH, Stockholm, Sweden, pp. 3389–3393.

Kawahara, H., Masuda-Katsuse, I., Cheveigne, A.D., 1999. Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds. Speech Commun. 27 (3−4), 187–207.

Ling, Z.-H., Sun, X.-H., Dai, L.-R., Hu, Y., 2016. Modulation spectrum compensation for HMM-based speech synthesis using line spectral pairs. In: Proc. ICASSP, Shanghai, China, pp. 5595–5599.

McAuley, C.D.J., Puckette, M., 2018. Synthesizing audio with GANs. In: Proceedings of the ICLR Workshop, Vancouver, Canada.

Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J., Courville, A., Bengio, Y., 2017. SampleRNN: an unconditional end-to-end neural audio generation model. In: Proceedings of the ICLR, Toulon, France.

Mirza, M., Osindero, S., 2014. Conditional generative adversarial networks. arXiv:1411.1784.

Morise, M., Yokomori, F., Ozawa, K., 2016. WORLD: a vocoder-based high-quality speech synthesis system for real-time applications. IEICE Transactions on Information and Systems E99-D (7), 1877–1884.

Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., Kavukcuoglu, K., 2016. WaveNet: a generative model for raw audio. arXiv:1609.03499.

Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., Driessche, G., Lockhart, E., Cobo, L., Stimberg, F., Casagrande, N., Grewe, D., Noury, S., Dieleman, S., Elsen, E., Kalchbrenner, N., Zen, H., Graves, A., King, H., Walters, T., Belov, D., Hassabis, D., 2017. Parallel WaveNet: fast high-fidelity speech synthesis. arXiv:1711.10433.

Sagisaka, Y., 1988. Speech synthesis by rule using an optimal selection of non-uniform synthesis units. In: Proceedings of the ICASSP, New York, USA, pp. 679–682. doi: 10.1109/ICASSP.1988.196677.

Sahidullah, M., Kinnunen, T., Hanilçi, C., 2015. A comparison of features for synthetic speech detection. In: Proceedings of the INTERSPEECH, Dresden, Germany, pp. 2087–2091.

Saito, Y., Takamichi, S., Saruwatari, H., 2017. Training algorithm to deceive anti-spoofing verification for DNN-based speech synthesis. In: Proceedings of the ICASSP, New Orleans, USA, pp. 4900–4904.

Saito, Y., Takamichi, S., Saruwatari, H., 2018a. Statistical parametric speech synthesis incorporating generative adversarial networks. IEEE/ACM Trans. Audio Speech Lang. Process. 26 (1), 84–96.

Saito, Y., Takamichi, S., Saruwatari, H., 2018b. Text-to-speech synthesis using STFT spectra based on low-/multi-resolution generative adversarial networks. In: Proceedings of the ICASSP, Calgary, Canada, pp. 5299–5303.

Sotelo, J., Mehri, S., Kumar, K., Santos, J.F., Kastner, K., Courville, A., Bengio, Y., 2017. Char2Wav: end-to-end speech synthesis. In: Proceedings of the ICLR, Toulon, France.

Takaki, S., Kameoka, H., Yamagishi, J., 2017. Direct modeling of frequency spectra and waveform generation based on phase recovery for DNN-based speech synthesis. In: Proceedings of the INTERSPEECH, Stockholm, Sweden, pp. 1128–1132.

Takamichi, S., Toda, T., Black, A.W., Neubig, G., Sakti, S., Nakamura, S., 2016. Postfilters to modify the modulation spectrum for statistical parametric speech synthesis. IEEE/ACM Trans. Audio, Speech, and Lang. Process. 24 (4), 755–767.

Toda, T., Black, A.W., Tokuda, K., 2007. Voice conversion based on maximum likelihood estimation of spectral parameter trajectory. IEEE Trans. Audio, Speech, and Lang. Process. 15 (8), 2222–2235.

Toda, T., Chen, L.H., Saito, D., Villavicencio, F., Wester, M., Wu, Z., Yamagishi, J., 2016. The voice conversion challenge 2016. In: Proceedings of the INTERSPEECH, California, USA, pp. 1632–1636.

Tokuda, K., Nankaku, Y., Toda, T., Zen, H., Yamagishi, J., Oura, K., 2013. Speech synthesis based on hidden Markov models. Proc. IEEE 101 (5), 1234–1252.

Wang, D., Chen, J., 2018. Supervised speech separation based on deep learning: an overview. IEEE/ACM Trans. Audio, Speech, and Lang. Process. 26 (10), 1702–1726.

Wu, Z., King, S., 2016. Improving trajectory modeling for DNN-based speech synthesis by using stacked bottleneck features and minimum trajectory error training. IEEE/ACM Trans. Audio, Speech, and Lang. Process. 24 (7), 1255–1265.

Wu, Z., Leon, P.L.D., Demiroglu, C., Khodabakhsh, A., King, S., Ling, Z., Saito, D., Stewart, B., Toda, T., Wester, M., Yamagishi, J., 2016. Antispoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance. IEEE/ACM Trans. Audio, Speech, and Lang. Process. 24 (4), 768–783.

Xu, Y., Du, J., Dai, L.-R., Lee, C.-H., 2015. A regression approach to speech enhancement based on deep neural networks. IEEE/ACM Trans. Audio, Speech, and Lang. Process. 23 (1), 7–19.

Zen, H., Sak, H., 2015. Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. In: Proceedings of the ICASSP, Brisbane, Australia, pp. 4470–4474.

Zen, H., Senior, A., Schuster, M., 2013. Statistical parametric speech synthesis using deep neural networks. In: Proceedings of the ICASSP, Vancouver, Canada, pp. 7962–7966.

Zen, H., Toda, T., 2005. An overview of nitech HMM-based speech synthesis system for blizzard challenge 2005. In: Proceedings of the INTERSPEECH, Lisbon, Portugal, pp. 93–96.

Zen, H., Tokuda, K., Black, A., 2009. Statistical parametric speech synthesis. Speech Commun. 51 (11), 1039–1064.