Statistical Parametric Speech Synthesis Incorporating Generative Adversarial Networks

Yuki Saito, Shinnosuke Takamichi, *Member, IEEE*, and Hiroshi Saruwatari, *Member, IEEE*

***Abstract*—A method for statistical parametric speech synthesis incorporating generative adversarial networks (GANs) is proposed. Although powerful deep neural networks techniques can be applied to artificially synthesize speech waveform, the synthetic speech quality is low compared with that of natural speech. One of the issues causing the quality degradation is an oversmoothing effect often observed in the generated speech parameters. A GAN introduced in this paper consists of two neural networks: a discriminator to distinguish natural and generated samples, and a generator to deceive the discriminator. In the proposed framework incorporating the GANs, the discriminator is trained to distinguish natural and generated speech parameters, while the acoustic models are trained to minimize the weighted sum of the conventional minimum generation loss and an adversarial loss for deceiving the discriminator. Since the objective of the GANs is to minimize the divergence (i.e., distribution difference) between the natural and generated speech parameters, the proposed method effectively alleviates the oversmoothing effect on the generated speech parameters. We evaluated the effectiveness for text-to-speech and voice conversion, and found that the proposed method can generate more natural spectral parameters and F_0 than conventional minimum generation error training algorithm regardless of its hyperparameter settings. Furthermore, we investigated the effect of the divergence of various GANs, and found that a Wasserstein GAN minimizing the Earth-Mover's distance works the best in terms of improving the synthetic speech quality.**

***Index Terms*—Statistical parametric speech synthesis, text-to-speech synthesis, voice conversion, deep neural networks, generative adversarial networks, over-smoothing.**

I. Introduction

STATISTICAL parametric speech synthesis (SPSS) [1] is a technique that aims to generate natural-sounding synthetic speech. Text-to-speech (TTS) synthesis [2] is a technique for synthesizing speech from text, and voice conversion (VC) [3] is a technique for synthesizing speech from another one while preserving linguistic information of original speech. In SPSS, acoustic models represent the relationship between input features and acoustic features. Recently, deep neural networks (DNNs) [4] have been utilized as the acoustic models for TTS and VC because they can model the relationship between input features and acoustic features more accurately than conventional hidden Markov models [5] and Gaussian mixture models [6]. These acoustic models are trained with several training algorithms such as the minimum generation error (MGE) criterion [7], [8]. Techniques for training the acoustic models to generate high-quality speech are widely studied since they can be used for both TTS and VC. However, the speech parameters generated from these models tend to be over-smoothed, and the resultant quality of speech is still low compared with that of natural speech [1], [9]. The over-smoothing effect is a common issue in both TTS and VC.

One way to improve speech quality is to reduce the difference between natural and generated speech parameters. For instance, since the parameter distributions of natural and synthetic speech are significantly different [10], we can improve the synthetic speech quality by transforming the generated speech parameters so that their distribution is close to that of natural speech. This can be done by, for example, modeling the probability distributions in a parametric [6] or non-parametric [11] way in the training stage, and then, generating or transforming the synthetic speech parameters by using the distributions. The more effective approach is to use analytically derived features correlated to the quality degradation of the synthetic speech. Global variance (GV) [6] and modulation spectrum (MS) [12] are well-known examples for reproducing natural statistics. These features work as a constraint in the training/synthesis stage [13], [14]. Nose and Ito [15] and Takamichi *et al.* [13] proposed methods that reduce the difference between the Gaussian distributions of natural and generated GV and MS. However, quality degradation is still a critical problem.

In order to address this quality problem, in this paper we propose a novel method using generative adversarial networks (GANs) for training acoustic models in SPSS. A GAN consists of two neural networks: a discriminator to distinguish natural and generated samples, and a generator to deceive the discriminator. Based on the framework, we define a new training criterion for the acoustic models; the criterion is the weighted sum of the conventional MGE training and an adversarial loss. The adversarial loss makes the discriminator recognize the generated speech parameters as natural. Since the objective of the GANs is to minimize the divergence (i.e., the distribution difference) between the natural and generated speech parameters,

Manuscript received March 10, 2017; revised July 26, 2017 and September 17, 2017; accepted September 22, 2017. Date of publication October 9, 2017; date of current version November 29, 2017. This work was supported in part by the ImPACT Program of Council for Science, Technology, and Innovation (Cabinet Office, Government of Japan), in part by SECOM Science and Technology Foundation, and in part by JSPS KAKENHI under Grants 16H06681 and JP17H06101. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Junichi Yamagishi. *(Corresponding author: Yuki Saito.)*

The authors are with the Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan (e-mail: yuuki_saito@ipc.i.u-tokyo.ac.jp; shinnosuke_takamichi@ipc.i.u-tokyo.ac.jp; hiroshi_saruwatari@ipc.i.u-tokyo.ac.jp).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2017.2761547

our method effectively alleviates the effect of over-smoothing the generated speech parameters. Moreover, our method can be regarded as a generalization of the conventional method using explicit modeling of analytically derived features such as GV and MS because it effectively minimizes the divergence without explicit statistical modeling. Also, the discriminator used in our method can be interpreted as anti-spoofing, namely, a technique for detecting synthetic speech and preventing voice spoofing attack. Accordingly, techniques and ideas concerning anti-spoofing can be applied to the training. We evaluated the effectiveness of the proposed method in DNN-based TTS and VC, and found that the proposed algorithm generates more natural spectral parameters and F_0 than those of the conventional MGE training algorithm and improves the synthetic speech quality regardless of its hyper-parameter settings which control the weight of the adversarial loss. Furthermore, we investigated the effect of the divergence of various GANs, including image-processing-related ones such as the least squares GAN (LS-GAN) and the Wasserstein GAN (W-GAN), and speech-processing-related ones such as the f-divergence GAN (f-GAN). The results of the investigation demonstrate that the W-GAN minimizing the Earth-Mover's distance works the best in regard to improving synthetic speech quality.

In Section II of this paper, we briefly review conventional training algorithms in DNN-based TTS and VC. Section III introduces GANs and proposes a method for speech synthesis incorporating those GANs. Section IV presents the experimental evaluations. We conclude in Section V with a summary.

II. Conventional DNN-Based SPSS

This section describes the conventional training algorithm for DNN-based SPSS, including TTS and VC.

A. DNN-Based TTS

1) DNNs as Acoustic Models: In DNN-based TTS [16], acoustic models representing the relationship between linguistic features and speech parameters consist of layered hierarchical networks. In training the models, we minimize the loss function calculated using the speech parameters of natural and synthetic speech. Let $\boldsymbol{x} = [\boldsymbol{x}_1^\top, \ldots, \boldsymbol{x}_t^\top, \ldots, \boldsymbol{x}_T^\top]^\top$ be a linguistic feature sequence, $\boldsymbol{y} = [\boldsymbol{y}_1^\top, \ldots, \boldsymbol{y}_t^\top, \ldots, \boldsymbol{y}_T^\top]^\top$ be a natural speech parameter sequence, and $\hat{\boldsymbol{y}} = [\hat{\boldsymbol{y}}_1^\top, \ldots, \hat{\boldsymbol{y}}_t^\top, \ldots, \hat{\boldsymbol{y}}_T^\top]^\top$ be a generated speech parameter sequence, where t and T denote the frame index and total frame length, respectively. $\boldsymbol{x}_t$ and $\boldsymbol{y}_t = [y_t\left(1\right), \ldots, y_t\left(D\right)]^\top$ are a linguistic parameter vector and a D-dimensional speech parameter vector at frame t, respectively.

2) Acoustic Model Training: The DNNs that predict a natural static-dynamic speech feature sequence $\boldsymbol{Y} = [\boldsymbol{Y}_1^\top, \ldots, \boldsymbol{Y}_t^\top, \ldots, \boldsymbol{Y}_T^\top]^\top$ from $\boldsymbol{x}$ are trained to minimize a defined training criterion. $\boldsymbol{Y}_t = [\boldsymbol{y}_t^\top, \Delta\boldsymbol{y}_t^\top, \Delta\Delta\boldsymbol{y}_t^\top]^\top$ is a natural static-dynamic speech feature at frame t. Given a predicted static-dynamic speech feature sequence $\hat{\boldsymbol{Y}} = [\hat{\boldsymbol{Y}}_1^\top, \ldots, \hat{\boldsymbol{Y}}_t^\top, \ldots, \hat{\boldsymbol{Y}}_T^\top]^\top$, the most standard criterion is the mean squared error (MSE) $L_{\mathrm{MSE}}(\boldsymbol{Y}, \hat{\boldsymbol{Y}})$ between $\boldsymbol{Y}$ and $\hat{\boldsymbol{Y}}$ defined as follows:

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

A set of the model parameters θ_{G} (e.g., weight and bias of DNNs) is updated by the backpropagation algorithm using the gradient $\nabla_{\theta_{\mathrm{G}}} L_{\mathrm{MSE}}(\boldsymbol{Y}, \hat{\boldsymbol{Y}})$.

To take the static-dynamic constraint into account, the minimum generation error (MGE) training algorithm was proposed [8]. In MGE training, the loss function $L_{\mathrm{MGE}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is defined as the mean squared error between natural and generated speech parameters as follows:

$$\begin{aligned} L_{\mathrm{MGE}}\left(\boldsymbol{y}, \hat{\boldsymbol{y}}\right) &= \frac{1}{T}\left(\hat{\boldsymbol{y}} - \boldsymbol{y}\right)^\top \left(\hat{\boldsymbol{y}} - \boldsymbol{y}\right) \\ &= \frac{1}{T}\left(\boldsymbol{R}\hat{\boldsymbol{Y}} - \boldsymbol{y}\right)^\top \left(\boldsymbol{R}\hat{\boldsymbol{Y}} - \boldsymbol{y}\right). \end{aligned} \tag{2}$$

$\boldsymbol{R}$ is a DT-by-$3DT$ matrix given as

$$\boldsymbol{R} = \left(\boldsymbol{W}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{W}\right)^{-1} \boldsymbol{W}^\top \boldsymbol{\Sigma}^{-1}, \tag{3}$$

where $\boldsymbol{W}$ is a $3DT$-by-DT matrix for calculating dynamic features [5] and $\boldsymbol{\Sigma} = \mathrm{diag}[\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_t, \ldots, \boldsymbol{\Sigma}_T]$ is a $3DT$-by-$3DT$ covariance matrix, where $\boldsymbol{\Sigma}_t$ is a $3D$-by-$3D$ covariance matrix at frame t. $\boldsymbol{\Sigma}$ is separately estimated using training data. We define the speech parameter prediction as $\hat{\boldsymbol{y}} = \boldsymbol{R}\hat{\boldsymbol{Y}} = \boldsymbol{G}(\boldsymbol{x}; \theta_{\mathrm{G}})$, where θ_{G} denotes the acoustic model parameters and it is updated by the backpropagation algorithm using the gradient of the generation error, $\nabla_{\theta_{\mathrm{G}}} L_{\mathrm{MGE}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$. As described in [8], the gradient includes $\nabla_{\hat{Y}} L_{\mathrm{MGE}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ given as $\boldsymbol{R}^\top(\hat{\boldsymbol{y}} - \boldsymbol{y})/T$.

Phoneme duration is predicted in the same manner without dynamic feature calculation. Let $\boldsymbol{d} = [d_1, \ldots, d_p, \ldots, d_P]^\top$ be a natural phoneme duration sequence, and $\hat{\boldsymbol{d}} = [\hat{d}_1, \ldots, \hat{d}_p, \ldots, \hat{d}_P]^\top$ be a duration sequence generated using duration models described as DNNs. p is the phoneme index and P is the total number of phonemes. The model parameters are updated to minimize $L_{\mathrm{MSE}}(\boldsymbol{d}, \hat{\boldsymbol{d}})$.

B. DNN-Based VC

DNN-based acoustic models for VC convert input speech features to desired output speech features. In training, a dynamic time warping algorithm is used to temporally align source and target speech features. Using the aligned features, $\boldsymbol{x}$ and $\boldsymbol{y}$, the acoustic models are trained to minimize $L_{\mathrm{MGE}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$, the same as DNN-based TTS.

III. DNN-Based SPSS Incorporating GAN

A. Generative Adversarial Networks (GANs) [17]

A GAN is a framework for learning deep generative models, which simultaneously trains two DNNs: a generator and discriminator $D(\boldsymbol{y}; \theta_{\mathrm{D}})$. θ_{D} is a set of the model parameters of the discriminator. The value obtained by taking the sigmoid function from the discriminator's output, $1/(1 + \exp(-D(\boldsymbol{y})))$, represents the posterior probability that input $\boldsymbol{y}$ is natural data. The discriminator is trained to make the posterior probability 1 for natural data and 0 for generated data, while the generator

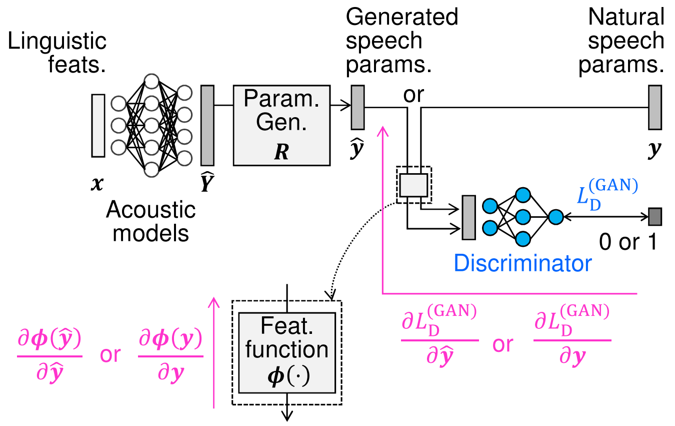
Linguistic feats.
Generated speech params.
Natural speech params.
Param. Gen. $\boldsymbol{R}$
$\boldsymbol{x}$
$\hat{\boldsymbol{Y}}$
$\hat{\boldsymbol{y}}$
or
$\boldsymbol{y}$
Acoustic models
$L_{\mathrm{D}}^{(\mathrm{GAN})}$
0 or 1
Discriminator
Feat. function $\boldsymbol{\phi}(\cdot)$
$\frac{\partial \boldsymbol{\phi}(\hat{\boldsymbol{y}})}{\partial \hat{\boldsymbol{y}}}$ or $\frac{\partial \boldsymbol{\phi}(\boldsymbol{y})}{\partial \boldsymbol{y}}$
$\frac{\partial L_{\mathrm{D}}^{(\mathrm{GAN})}}{\partial \hat{\boldsymbol{y}}}$ or $\frac{\partial L_{\mathrm{D}}^{(\mathrm{GAN})}}{\partial \boldsymbol{y}}$

Fig. 1. Loss function and gradients for updating the discriminator. Param. Gen. indicates the speech parameter generation [5]. Note that, the model parameters of the acoustic models are not updated in this step.

is trained to deceive the discriminator; that is, it tries to make the discriminator make the posterior probability 1 for generated data.

In the GAN training, the two DNNs are iteratively updated by minibatch stochastic gradient descent. First, by using natural data $\boldsymbol{y}$ and generated data $\hat{\boldsymbol{y}}$, we calculate the discriminator loss $L_{\mathrm{D}}^{(\mathrm{GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ defined as the following cross-entropy function:

$$L_{\mathrm{D}}^{(\mathrm{GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} \log \frac{1}{1+\exp(-D(\boldsymbol{y}_t))} - \frac{1}{T}\sum_{t=1}^{T} \log\left(1 - \frac{1}{1+\exp(-D(\hat{\boldsymbol{y}}_t))}\right). \quad (4)$$

θ_{D} is updated by using the stochastic gradient $\nabla_{\theta_{\mathrm{D}}} L_{\mathrm{D}}^{(\mathrm{GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$. Fig. 1 illustrates the procedure for computing the discriminator loss. After updating the discriminator, we calculate the adversarial loss of the generator $L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}})$ which deceives the discriminator as follows:

$$L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} \log \frac{1}{1+\exp(-D(\hat{\boldsymbol{y}}_t))}. \quad (5)$$

A set of the model parameters of the generator θ_{G} is updated by using the stochastic gradient $\nabla_{\theta_{\mathrm{G}}} L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}})$. Goodfellow *et al.* [17] showed this adversarial framework minimizes the approximated Jensen–Shannon (JS) divergence between two distributions of natural and generated data.

B. Acoustic Model Training Incorporating GAN

Here, we describe a novel training algorithm for SPSS which incorporates the GAN. As for the proposed algorithm, acoustic models are trained to deceive the discriminator that distinguishes natural and generated speech parameters.

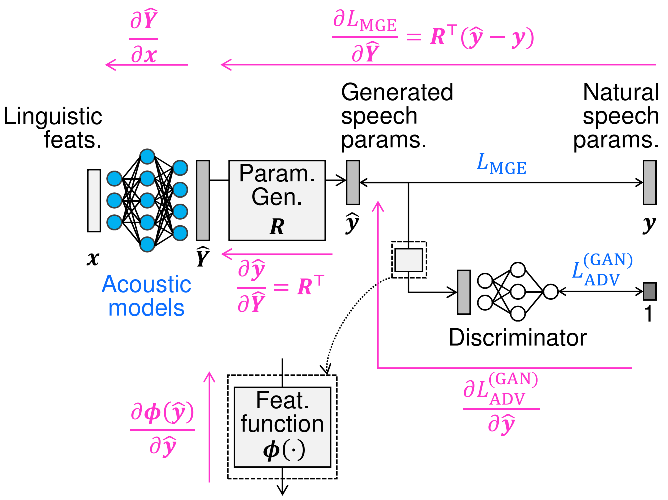
$\frac{\partial \hat{\boldsymbol{Y}}}{\partial \boldsymbol{x}}$
$\frac{\partial L_{\mathrm{MGE}}}{\partial \hat{\boldsymbol{Y}}} = \boldsymbol{R}^{\top}(\hat{\boldsymbol{y}} - \boldsymbol{y})$
Linguistic feats.
Generated speech params.
Natural speech params.
Param. Gen. $\boldsymbol{R}$
L_{MGE}
$\boldsymbol{x}$
$\hat{\boldsymbol{Y}}$
$\hat{\boldsymbol{y}}$
$\boldsymbol{y}$
Acoustic models
$\frac{\partial \hat{\boldsymbol{y}}}{\partial \hat{\boldsymbol{Y}}} = \boldsymbol{R}^{\top}$
$L_{\mathrm{ADV}}^{(\mathrm{GAN})}$
1
Discriminator
Feat. function $\boldsymbol{\phi}(\cdot)$
$\frac{\partial \boldsymbol{\phi}(\hat{\boldsymbol{y}})}{\partial \hat{\boldsymbol{y}}}$
$\frac{\partial L_{\mathrm{ADV}}^{(\mathrm{GAN})}}{\partial \hat{\boldsymbol{y}}}$

Fig. 2. Loss functions and gradients for updating acoustic models in the proposed method. Note that the model parameters of the discriminator are not updated in this step.

The loss function of speech synthesis is defined as the following:

$$L_{\mathrm{G}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = L_{\mathrm{MGE}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) + \omega_{\mathrm{D}} \frac{E_{L_{\mathrm{MGE}}}}{E_{L_{\mathrm{ADV}}}} L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}}), \quad (6)$$

where $L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}})$ makes the discriminator recognize the generated speech parameters as natural, and minimizes the divergence between the distributions of the natural and generated speech parameters. Therefore, the proposed loss function not only minimizes the generation error but also makes the distribution of the generated speech parameters close to that of natural speech. $E_{L_{\mathrm{MGE}}}$ and $E_{L_{\mathrm{ADV}}}$ denote the expectation values of $L_{\mathrm{MGE}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ and $L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}})$, respectively. Their ratio $E_{L_{\mathrm{MGE}}}/E_{L_{\mathrm{ADV}}}$ is the scale normalization term between the two loss functions, and the hyper-parameter ω_{D} controls the weight of the second term. When $\omega_{\mathrm{D}} = 0$, the loss function is equivalent to the conventional MGE training, and when $\omega_{\mathrm{D}} = 1$, the two loss functions have equal weights. A set of the model parameters of the acoustic models θ_{G} is updated by using the stochastic gradient $L_{\mathrm{G}}(\vec{y}, \vec{\hat{y}})$. Fig. 2 illustrates the procedure for computing the proposed loss function. In our algorithm, the acoustic models and discriminator are iteratively optimized, as shown in Algorithm 1. When one module is being updated, the model parameters of the another are fixed; that is, although the discriminator is included in the forward path to calculate $L_{\mathrm{ADV}}^{(\mathrm{GAN})}(\hat{\boldsymbol{y}})$ in $L_{\mathrm{G}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$, θ_{D} is not updated by the backpropagation for the acoustic models.

The discriminator used in our method can be regarded as a DNN-based anti-spoofing (voice spoofing detection) [18], [19] that distinguishes natural and synthetic speech. From this perspective, a feature function $\phi(\cdot)$ can be inserted between speech parameter prediction and the discriminator as shown in Figs. 1 and 2. The function calculates more distinguishable features in anti-spoofing than the direct use of speech parameters them-

Algorithm 1: Iterative optimization for acoustic models and discriminator.

```
$\eta :=$ learning rate
for number of training iterations do
    for all training data $(\boldsymbol{x}, \boldsymbol{y})$ do
        generate $\hat{\boldsymbol{y}}$ from the acoustic models:
            $\hat{\boldsymbol{y}} = \boldsymbol{G}(\boldsymbol{x})$.
        update $\theta_{\mathrm{D}}$ while fixing $\theta_{\mathrm{G}}$:
            $\theta_{\mathrm{D}} \leftarrow \theta_{\mathrm{D}} - \eta \nabla_{\theta_{\mathrm{D}}} L_{\mathrm{D}}^{(\mathrm{GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$.
        update $\theta_{\mathrm{G}}$ while fixing $\theta_{\mathrm{D}}$:
            $\theta_{\mathrm{G}} \leftarrow \theta_{\mathrm{G}} - \eta \nabla_{\theta_{\mathrm{G}}} L_{\mathrm{G}}(\boldsymbol{y}, \hat{\boldsymbol{y}})$.
    end for
end for
```

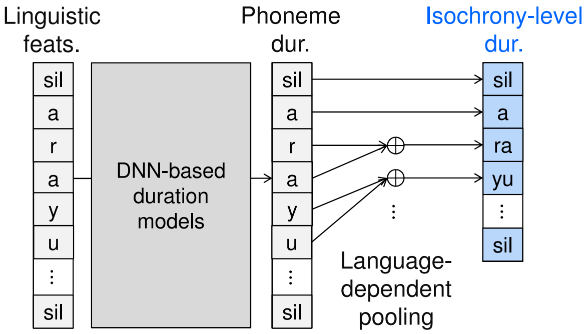

Fig. 3. Architecture to calculate isochrony-level duration from phoneme duration. In the case of Japanese, which has mora isochrony, each mora duration is calculated from the corresponding phoneme duration, e.g., the mora duration of /ra/ is calculated as the sum of the phoneme durations of /r/ and /a/.

selves. Namely, instead of $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ in (4) and (5), $\boldsymbol{\phi}(\boldsymbol{y})$ and $\boldsymbol{\phi}(\hat{\boldsymbol{y}})$ are used. In training the acoustic models, the gradient $\partial\boldsymbol{\phi}(\hat{\boldsymbol{y}})/\partial\hat{\boldsymbol{y}}$ is used for backpropagation. For example, when $\boldsymbol{\phi}(\hat{\boldsymbol{y}}) = \boldsymbol{W}\hat{\boldsymbol{y}}$, the gradient $\boldsymbol{W}^{\top}$ is used for backpropagation.

C. Application to F_0 and Duration Generation

Our algorithm is simply applied to the spectral parameter generation and conversion for TTS and VC. Here, we extend our algorithm to F_0 and duration generation in TTS. For F_0 generation, we use a continuous F_0 sequence [20] instead of the F_0 sequence because of the simple implementation. The input of the discriminator is the joint vector of a spectral parameter vector and continuous F_0 value of each frame.

For duration generation, although we can directly apply our algorithm to phoneme duration, it is not guaranteed that naturally-distributed phoneme duration has natural isochrony of the target language (e.g., moras in Japanese) [21]. Therefore, we modify our algorithm so that the generated duration naturally distributes in the language-dependent isochrony level. Fig. 3 shows the architecture. In the case of Japanese, which has mora isochrony, each mora duration is calculated from the corresponding phoneme durations.

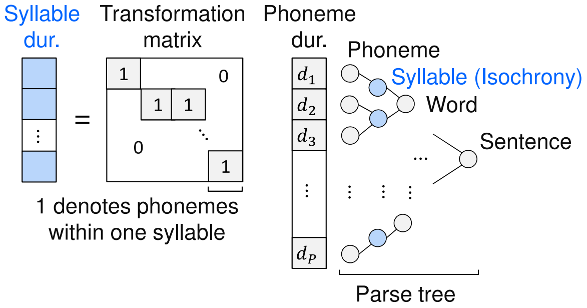

Fig. 4. Matrix representation to calculate isochrony-level duration. This is an example in the case of a syllable-timed language such as Chinese.

The discriminator minimizes the cross-entropy function by using the isochrony-level duration, while the generator minimizes the weighted sum of the MSE between natural and generated phoneme durations and the adversarial loss using the isochrony-level durations. Since the calculation of the isochrony-level duration is represented as the matrix multiplication shown in Fig. 4, the backpropagation is done using the transpose of the transformation matrix.

D. GANs to be Applied to the Proposed Method

The GAN framework works as a divergence minimization between natural and generated speech parameters. As described in Section III-B, the original GAN [17] minimizes the approximated JS divergence. From the perspective of the divergence minimization, we further introduce additional GANs minimizing other divergences: f-GAN [22], Wasserstein GAN (W-GAN) [23], and least squares GAN (LS-GAN) [24]. The divergence of the f-GAN is strongly related to speech processing such as a nonnegative matrix factorization [25], [26], and the effectiveness of the W-GAN and LS-GAN in the image processing is known. The discriminator loss $L_{\mathrm{D}}^{(*\text{-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ and adversarial loss $L_{\mathrm{ADV}}^{(*\text{-GAN})}(\hat{\boldsymbol{y}})$ introduced below can be used instead of (4) and (5), respectively.

1) f-GAN [22]: The f-GAN is the unified framework that encompasses the original GAN. The difference between distributions of natural and generated data is defined as the f-divergence [27], which is a large class of different divergences including the Kullback–Leibler (KL) and JS divergence. The f-divergence $\mathcal{D}_f(\boldsymbol{y}\|\hat{\boldsymbol{y}})$ is defined as follows:

$$\mathcal{D}_f\left(\boldsymbol{y}\|\hat{\boldsymbol{y}}\right) = \int q\left(\hat{\boldsymbol{y}}\right) f\left(\frac{p\left(\boldsymbol{y}\right)}{q\left(\hat{\boldsymbol{y}}\right)}\right) d\boldsymbol{y}, \tag{7}$$

where $p(\cdot)$ and $q(\cdot)$ are absolutely continuous density functions of $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$, respectively. $f(\cdot)$ is a convex function satisfying $f(1) = 0$. Although various choices of $f(\cdot)$ for recovering popular divergences are available, we adopt ones related to speech processing.

KL-GAN: Defining $f(r) = r\log r$ gives the KL divergence as follows:

$$\mathcal{D}_{\mathrm{KL}}\left(\boldsymbol{y}\|\hat{\boldsymbol{y}}\right) = \int p\left(\boldsymbol{y}\right) \log \frac{p\left(\boldsymbol{y}\right)}{q\left(\hat{\boldsymbol{y}}\right)} d\boldsymbol{y}. \tag{8}$$

The discriminator loss $L_{\mathrm{D}}^{(\text{KL-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{D}}^{(\text{KL-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} D(\boldsymbol{y}_t) + \frac{1}{T}\sum_{t=1}^{T} \exp(D(\hat{\boldsymbol{y}}_t) - 1), \tag{9}$$

while the adversarial loss $L_{\mathrm{ADV}}^{(\text{KL-GAN})}(\hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{ADV}}^{(\text{KL-GAN})}(\hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} D(\hat{\boldsymbol{y}}_t). \tag{10}$$

Reversed KL (RKL)-GAN: Since the KL divergence is not symmetric, the reversed version, called reversed KL (RKL) divergence $\mathcal{D}_{\mathrm{RKL}}(\boldsymbol{y}\|\hat{\boldsymbol{y}})$ differs from $\mathcal{D}_{\mathrm{KL}}(\boldsymbol{y}\|\hat{\boldsymbol{y}})$, which is defined as follows:

$$\mathcal{D}_{\mathrm{RKL}}(\boldsymbol{y}\|\hat{\boldsymbol{y}}) = \int q(\hat{\boldsymbol{y}}) \log \frac{q(\hat{\boldsymbol{y}})}{p(\boldsymbol{y})} d\boldsymbol{y} = \mathcal{D}_{\mathrm{KL}}(\hat{\boldsymbol{y}}\|\boldsymbol{y}). \tag{11}$$

Defining $f(r) = -\log r$ gives the discriminator loss $L_{\mathrm{D}}^{(\text{RKL-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ as follows:

$$L_{\mathrm{D}}^{(\text{RKL-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{T}\sum_{t=1}^{T} \exp(-D(\boldsymbol{y}_t)) + \frac{1}{T}\sum_{t=1}^{T} (-1 + D(\hat{\boldsymbol{y}}_t)), \tag{12}$$

while the adversarial loss $L_{\mathrm{ADV}}^{(\text{RKL-GAN})}(\hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{ADV}}^{(\text{RKL-GAN})}(\hat{\boldsymbol{y}}) = \frac{1}{T}\sum_{t=1}^{T} \exp(-D(\hat{\boldsymbol{y}}_t)). \tag{13}$$

JS-GAN: The JS divergence without approximation can be formed within the f-GAN framework. Defining $f(r) = -(r + 1)\log\frac{r+1}{2} + r\log r$ gives the JS divergence as follows:

$$\mathcal{D}_{\mathrm{JS}}(\boldsymbol{y}\|\hat{\boldsymbol{y}}) = \frac{1}{2}\int p(\boldsymbol{y}) \log \frac{2p(\boldsymbol{y})}{p(\boldsymbol{y}) + q(\hat{\boldsymbol{y}})} d\boldsymbol{y} + \frac{1}{2}\int q(\hat{\boldsymbol{y}}) \log \frac{2q(\hat{\boldsymbol{y}})}{p(\boldsymbol{y}) + q(\hat{\boldsymbol{y}})} d\boldsymbol{y}. \tag{14}$$

The discriminator loss $L_{\mathrm{D}}^{(\text{JS-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{D}}^{(\text{JS-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} \log \frac{2}{1 + \exp(-D(\boldsymbol{y}_t))} - \frac{1}{T}\sum_{t=1}^{T} \log\left(2 - \frac{2}{1 + \exp(-D(\hat{\boldsymbol{y}}_t))}\right), \tag{15}$$

while the adversarial loss $L_{\mathrm{ADV}}^{(\text{JS-GAN})}(\hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{ADV}}^{(\text{JS-GAN})}(\hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} \log \frac{2}{1 + \exp(-D(\hat{\boldsymbol{y}}_t))}. \tag{16}$$

Note that, the approximated JS divergence minimized by the original GAN is $2\mathcal{D}_{\mathrm{JS}}(\boldsymbol{y}\|\hat{\boldsymbol{y}}) - \log(4)$ [17].

2) Wasserstein GAN (W-GAN) [23]: To stabilize the extremely unstable training of the original GAN, Arjovsky *et al.* [23] proposed the W-GAN, which minimizes the Earth-Mover's distance (Wasserstein-1). The Earth-Mover's distance is defined as follows:

$$\mathcal{D}_{\mathrm{EM}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \inf_{\gamma} \mathbb{E}_{(\boldsymbol{y},\hat{\boldsymbol{y}})\sim\gamma}[\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|], \tag{17}$$

where $\gamma(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is the joint distribution whose marginals are respectively the distributions of $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$. On the basis of the Kantorovich–Rubinstein duality [28], the discriminator loss $L_{\mathrm{D}}^{(\text{W-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{D}}^{(\text{W-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} D(\boldsymbol{y}_t) + \frac{1}{T}\sum_{t=1}^{T} D(\hat{\boldsymbol{y}}_t), \tag{18}$$

while the adversarial loss $L_{\mathrm{ADV}}^{(\text{W-GAN})}(\hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{ADV}}^{(\text{W-GAN})}(\hat{\boldsymbol{y}}) = -\frac{1}{T}\sum_{t=1}^{T} D(\hat{\boldsymbol{y}}_t). \tag{19}$$

We assume the discriminator to be the K-Lipschitz function. Namely, after updating the discriminator, we clamp its weight parameters to a fixed interval such as $[-0.01, 0.01]$.

3) Least Squares GAN (LS-GAN) [24]: To avoid the gradient vanishing problem of the original GAN using the sigmoid cross entropy, Mao *et al.* [24] proposed the LS-GAN, which formulates the objective function minimizing the mean squared error. The discriminator loss $L_{\mathrm{D}}^{(\text{LS-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{D}}^{(\text{LS-GAN})}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{2T}\sum_{t=1}^{T} (D(\boldsymbol{y}_t) - b)^2 + \frac{1}{2T}\sum_{t=1}^{T} (D(\hat{\boldsymbol{y}}_t) - a)^2, \tag{20}$$

while the adversarial loss $L_{\mathrm{ADV}}^{(\text{LS-GAN})}(\hat{\boldsymbol{y}})$ is defined as follows:

$$L_{\mathrm{ADV}}^{(\text{LS-GAN})}(\hat{\boldsymbol{y}}) = \frac{1}{2T}\sum_{t=1}^{T} (D(\hat{\boldsymbol{y}}_t) - c)^2, \tag{21}$$

where a, b, and c denote the labels that make the discriminator recognize the generated data as generated, the natural data as natura, and the generated data as natural, respectively. When they satisfy the conditions $b - c = 1$ and $b - a = 2$, the divergence to be minimized is the Pearson $\mathcal{X}^2$ divergence between $p(\boldsymbol{y}) + q(\hat{\boldsymbol{y}})$ and $2q(\hat{\boldsymbol{y}})$. Because we found that these conditions degrade quality of synthetic speech, we used alternative conditions suggested in [24, eq. (9)], i.e., $a = 0$, $b = 1$, and $c = 1$.

E. Discussions

The proposed loss function (6) is the combination of a multi-task learning algorithm using discriminators [29] and GANs. In defining $L_{\mathrm{G}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = L_{\mathrm{ADV}}^{(\text{GAN})}(\hat{\boldsymbol{y}})$, the loss function is equivalent to that for the GAN. Comparing with the GANs, our method

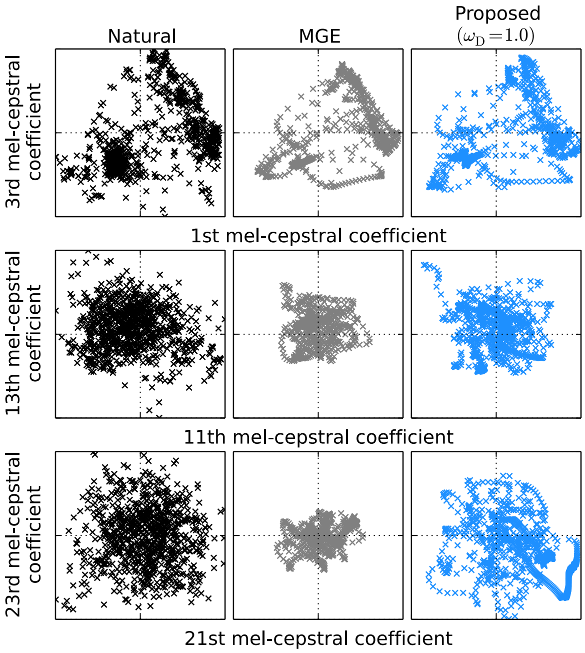

Fig. 5. Scatter plots of mel-cepstral coefficients with several pairs of dimensions. From the left, the figures correspond to natural speech, the conventional MGE algorithm, and the proposed algorithm ($\omega_D = 1.0$). These mel-cepstral coefficients were extracted from one utterance of the evaluation data.

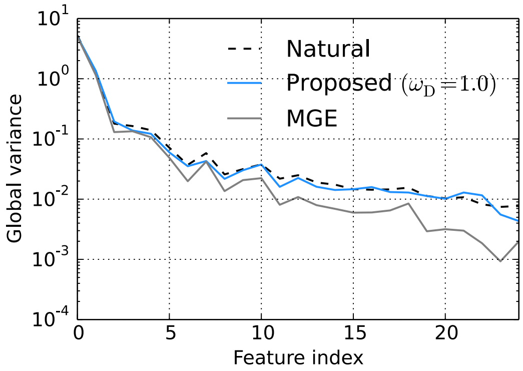

Fig. 6. Averaged GVs of mel-cepstral coefficients. Dashed, black, and blue lines correspond to natural speech, the conventional MGE, and the proposed algorithm, respectively.

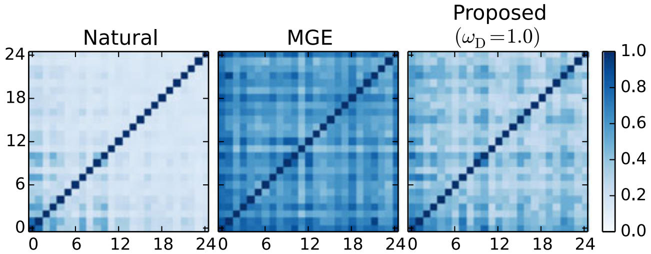

Fig. 7. MICs of natural and generated mel-cepstral coefficients. The MIC ranges from 0.0 to 1.0, and the two variables with a strong correlation have a value closer to 1.0. From the left, the figures correspond to natural speech, the conventional MGE algorithm, and the proposed algorithm ($\omega_D = 1.0$). These MICs were calculated from one utterance of the evaluation data.

TABLE I
STATISTICS OF NATURAL ("NATURAL") AND GENERATED ("MGE" AND "PROPOSED") CONTINUOUS F_0

	Mean	Variance
Natural	4.8784	0.076853
MGE	4.8388	0.032841
Proposed ($\omega_D = 1.0$)	**4.8410**	**0.032968**

is a fully supervised setting, i.e., we utilize the referred input and output parameters [30] without a latent variable. Also, since only the backpropagation algorithm is used for training, a variety of DNN architectures such as long short-term memory (LSTM) [31] can be used as the acoustic models and discriminator.

Using the designed feature function $\phi(\cdot)$, we can choose not only analytically derived features (e.g., GV and MS) but also automatically derived features (e.g., auto-encoded features [32]).

As described above, our algorithm makes the distribution of the generated speech parameters close to that of the natural speech. Since we perform generative adversarial training with DNNs, our algorithm comes to have a more complicated probability distribution than the conventional Gaussian distribution. Fig. 5 plots natural and generated speech parameters with several mel-cepstral coefficient pairs. Whereas the parameters of the conventional algorithm are narrowly distributed, those of the proposed algorithm are as widely distributed as the natural speech. Moreover, we can see that the proposed algorithm has a greater effect on the distribution of the higher order of the mel-cepstral coefficients.

Here, one can explore which components (e.g., analytically derived features and intuitive reasons [33]) the algorithm changes. Fig. 6 plots the averaged GVs of natural and generated speech parameters. We can see that the GV generated by the proposed algorithm is closer to the natural GV than that of the one produced by the conventional algorithm. This is quite natural result because compensating distribution differences is related to minimizing moments differences [34], [35]. Then, we calculated a maximal information coefficient (MIC) [36] to quantify a nonlinear correlation among the speech parameters. The results are shown in Fig. 7. As reported in [10], we can see that there are weak correlations among the natural speech parameters, whereas strong correlations are observed among those of the generated speech parameters of the MGE training. Moreover, the generated mel-cepstral coefficients of our algorithm have weaker correlations than those of the MGE training. These results suggest that the proposed algorithm compensates not only the GV of the generated speech parameters but also the correlation among the parameters. Also, the statistics of continuous F_0, phoneme duration, and mora duration are listed in Tables I, II, and III, respectively. The bold values are the closest to natural statistics in the results. In Tables II and III, "Proposed (phoneme)" and "Proposed (mora)" indicate that the proposed methods applied to phoneme and mora duration, respectively. We can see that the proposed method also makes the statistics closer to those of the natural speech than the conventional method. In the results

TABLE II
STATISTICS OF NATURAL ("NATURAL") AND GENERATED ("MSE" AND "PROPOSED (*)") PHONEME DURATION

	Mean	Variance
Natural	16.314	126.20
MSE	14.967	47.665
Proposed (phoneme, $\omega_{\mathrm{D}} = 1.0$)	14.963	**75.471**
Proposed (mora, $\omega_{\mathrm{D}} = 1.0$)	**15.074**	73.207

TABLE III
STATISTICS OF NATURAL ("NATURAL") AND GENERATED ("MSE" AND "PROPOSED (*)") MORA DURATION

	Mean	Variance
Natural	25.141	131.93
MSE	23.492	60.891
Proposed (phoneme, $\omega_{\mathrm{D}} = 1.0$)	24.794	**96.828**
Proposed (mora, $\omega_{\mathrm{D}} = 1.0$)	**24.978**	96.682

concerning duration generations, "Proposed (mora)," tends to reduce the difference in the mean rather than in the variance.

Our algorithm for spectrum and F_0, proposed in Section III-C, compensates the joint distribution of them. Therefore, we can perform the distribution compensation considering correlations [38] between different features. Also, compensating dimensionality differences [39] can be applied for deceiving the discriminator. Since the time resolutions in phoneme duration and mora duration are different, our algorithm considering isochrony is related to multi-resolution GAN [40] and hierarchical duration modeling [41].

Regarding related work, Kaneko *et al.* [42] proposed a generative adversarial network-based post-filter for TTS. The post-filtering process has high portability because it is independent of original speech synthesis procedures, but it comes at a high computation cost and has a heavy disk footprint in synthesis. In contrast, our algorithm can directly utilize original synthesis procedures [43]. Also, we expect that our algorithm can be extended to waveform synthesis [44], [45].

IV. EXPERIMENTAL EVALUATION

In this section, we evaluate the effectiveness of the proposed algorithm in terms of spectral parameters, F_0, and duration generation in DNN-based TTS, and then evaluate spectral parameter conversion in DNN-based VC.

A. *Experimental Conditions in TTS Evaluation*

We used speech data of a male speaker taken from the ATR Japanese speech database [46]. The speaker uttered 503 phonetically balanced sentences. We used 450 sentences (subsets A to I) for the training and 53 sentences (subset J) for the evaluation. Speech signals were sampled at a rate of 16 kHz, and the shift length was set to 5 ms. The 0th-through-24th mel-cepstral coefficients were used as spectral parameters and F_0 and 5 band-aperiodicity [47], [48] were used as excitation parameters. The STRAIGHT analysis-synthesis system [49] was used for the parameter extraction and the waveform synthesis. To improve training accuracy, speech parameter trajectory smoothing [50] with a 50 Hz cutoff modulation frequency was applied to the spectral parameters in the training data. In the training phase, spectral features were normalized to have zero-mean unit-variance, and 80% of the silent frames were removed from the training data in order to increase training accuracy.

The DNN architectures are listed in Table IV. In the spectral parameter generation (Sections IV-B1 and IV-B2), the acoustic models predicted static-dynamic feature sequence of the mel-cepstral coefficients (75-dim.) from the 274-dimensional linguistic features frame by frame, and the discriminator used frame-wise static mel-cepstral coefficients (25-dim.). Here, since F_0, band-aperiodicity, and duration of natural speech were directly used for the speech waveform synthesis, we only used some of the prosody-related features such as the accent type. In the spectral parameter and F_0 generation (Section IV-B3), the acoustic models predicted static-dynamic feature sequence of the mel-cepstral coefficients, continuous log F_0 [20], and band-aperiodicity with a voiced/unvoiced flag (94-dim.) from the 442-dimensional linguistic features frame by frame, and the discriminator used the joint vector of the frame-wise static mel-cepstral coefficients and continuous log F_0 (26-dim.). In the duration generation (Section IV-B3), we constructed duration models that generate phoneme duration from corresponding linguistic features (439-dim). The acoustic models were trained using MGE training.

In the training phase, we ran the training algorithm based on minimizing the MSE (1) [16] frame-by-frame for the initialization of acoustic models and then we ran the conventional MGE training [8] with 25 iterations. Here, "iteration" means using all the training data (450 utterances) once for training. The discriminator was initialized using natural speech and synthetic speech after the MGE training. The number of iterations for the discriminator initialization was 5. The proposed training and discriminator re-training were performed with 25 iterations. The expectation values $E_{L_{\mathrm{MGE}}}$ and $E_{L_{\mathrm{ADV}}}$ were estimated at each iteration step.

B. *Evaluation in TTS*

1) Objective Evaluation With Hyper-Parameter Settings: In order to evaluate our algorithm, we calculated the parameter generation loss defined in (2) and the spoofing rate of the synthetic speech. The spoofing rate is the number of spoofing synthetic speech parameters divided by the total number of synthetic speech parameters in the evaluation data. Here, "spoofing synthetic speech parameter" indicates a parameter for which the discriminator recognized the synthetic speech as natural. The discriminator for calculating the spoofing rates was constructed using natural speech parameters and generated speech parameters of the conventional MGE training. The generation loss and spoofing rates were first calculated with various hyper-parameter ω_{D} settings.

Fig. 8 shows the results for the generation loss and spoofing rate.

TABLE IV
ARCHITECTURES OF DNNs USED IN TTS EVALUATIONS

	Spectral parameter generation (Section IV-B1 and IV-B2)	Spectral and F_0 parameter generation (Section IV-B3)	Duration generation (Section IV-B4)
Acoustic models	274–3 × 400 (ReLU)–75 (linear)	442–3 × 512 (ReLU)–94 (linear)	442–3 × 512 (ReLU)–94 (linear)
Discriminator	25–2 × 200 (ReLU)–1 (sigmoid)	26–3 × 256 (ReLU)–1 (sigmoid)	1–3 × 256 (ReLU)–1 (sigmoid)
Duration models	N/A	439–3 × 256 (ReLU)–1 (linear)	439–3 × 256 (ReLU)–1 (linear)

Feed-forward networks were used for all architectures. ReLU indicates rectified linear unit [37].

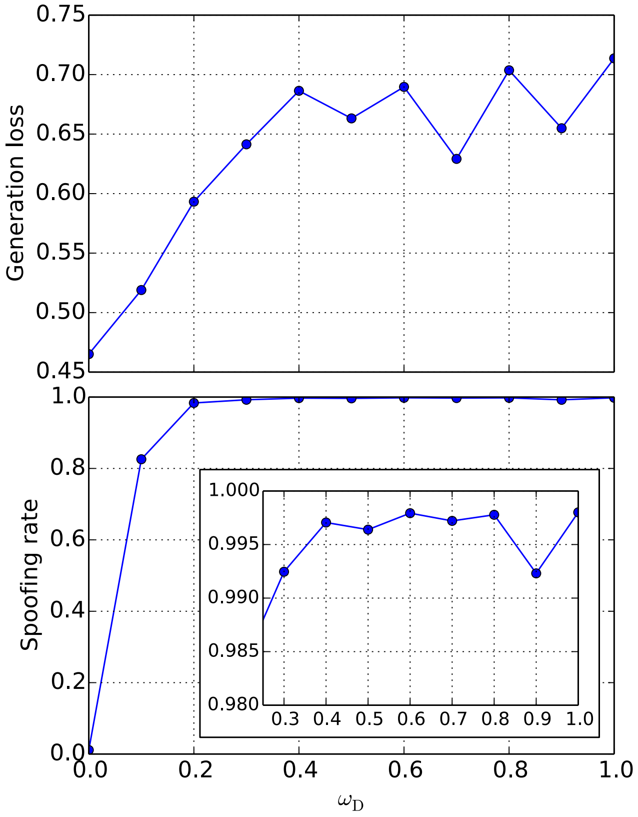

Fig. 8. Parameter generation loss (above) and spoofing rate (below) for various ω_D for spectral parameter generation in TTS.

As ω_D increases from 0.0, the generation loss monotonically increases, but from 0.4, we cannot see any tendency. On the other hand, the spoofing rate significantly increases as ω_D increases from 0.0 to 0.2; from 0.2, the value does not vary much. These results demonstrate that the proposed training algorithm makes the generation loss worse but can train the acoustic models to deceive the discriminator; in other words, although our method does not necessarily decrease the generation error, it tries to reduce the difference between the distributions of natural and generated speech parameters by taking the adversarial loss into account during the training.

2) Investigation of Convergence in Training: To investigate the convergence of the proposed training algorithm, we ran the algorithm through 100 iterations. Fig. 9 plots the generation loss and adversarial loss for the training and evaluation data. We can see that both loss values are almost monotonically decreased in training. Although the values of evaluation data strongly vary after a few iterations, they can converge after several more iterations.

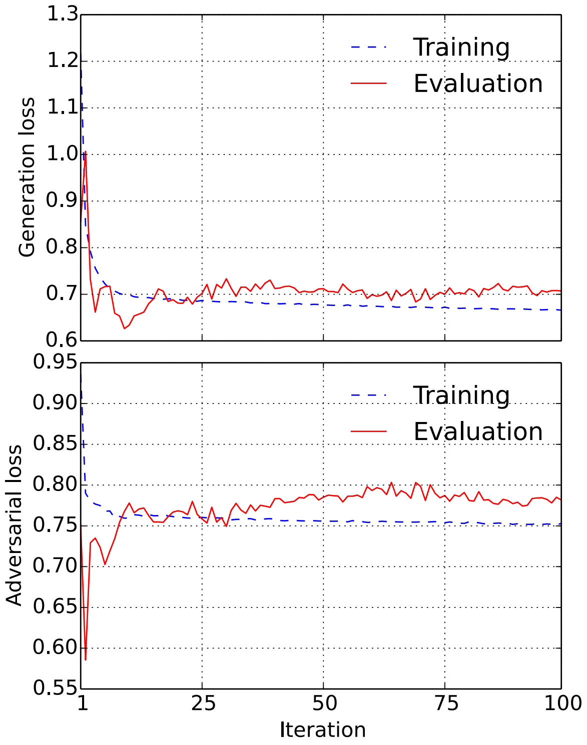

Fig. 9. Parameter generation loss (above) and adversarial loss (below) for the training data (blue-dashed line) and evaluation data (red line).

3) Subjective Evaluation of Spectral Parameter Generation: A preference test (AB) test was conducted to evaluate the quality of speech produced by the algorithm. We generated speech samples with three methods:

[MGE:] conventional MGE (= Proposed $(\omega_D = 0.0)$)
[Proposed $(\omega_D = 0.3)$:] spoofing rate > 0.99
[Proposed $(\omega_D = 1.0)$:] standard setting

Every pair of synthetic speech samples generated by using each method was presented to listeners in random order. Listeners participated in the assessment by using our crowdsourced subjective evaluation systems.

The results are shown in Fig. 10.

In Fig. 10(a) and (b), the proposed algorithm outperforms conventional MGE training algorithm in both hyper-parameter settings. Therefore, we can conclude that our algorithm robustly yields significant improvement in terms of speech quality regardless of its hyper-parameter setting. Henceforth, we set the

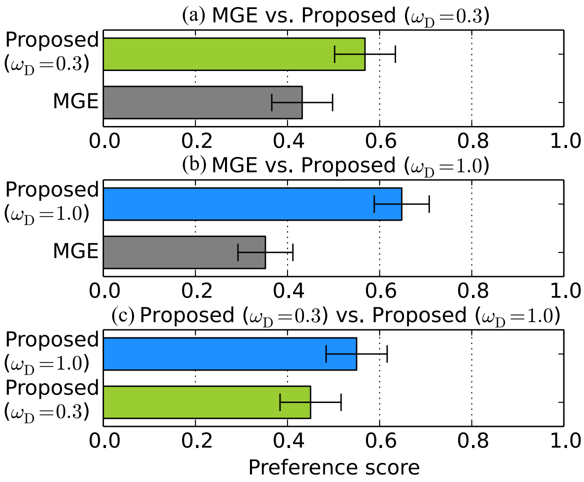

Fig. 10. Preference scores of speech quality with 95% confidence intervals (spectral parameter generation in TTS). From the top, the numbers of listeners were 22, 24, and 22, respectively.

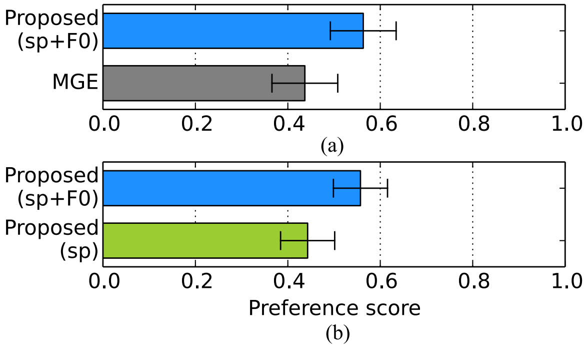

Fig. 11. Preference scores of speech quality with 95% confidence intervals (spectral parameter and F_0 generation in TTS). From the top, the numbers of the listeners were 19 and 28, respectively. (a) MGE vs. Proposed (sp+F0). (b) Proposed (sp) vs. Proposed (sp+F0).

hyper-parameter to 1.0 for the following evaluations because Fig. 10(c) shows that the score of "Proposed ($\omega_D = 1.0$)" was slightly better than that of "Proposed ($\omega_D = 0.3$)."

4) Subjective Evaluation of F_0 Generation: We evaluated the effect of the proposed algorithm for F0 generation. We conducted a subjective evaluation using the following three methods:

[MGE:] conventional MGE
[Proposed (sp):] proposed algorithm applied only to spectral parameters
[Proposed (sp+F0):] proposed algorithm applied to spectral and F_0 parameters

Every pair of synthetic speech samples generated by using each method was presented to listeners in random order. Since Fig. 10 has already demonstrated that the proposed algorithm improves synthetic speech quality in terms of generating spectral parameters, we did not compare "Proposed (sp)" with "MGE." Listeners participated in the assessment by using our crowdsourced subjective evaluation systems.

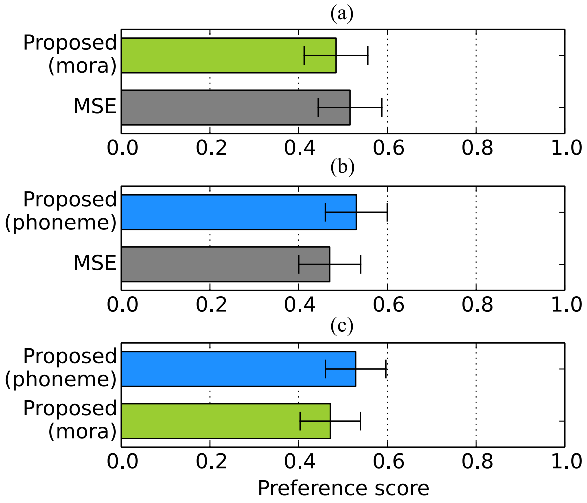

Fig. 12. Preference scores of speech quality with 95% confidence intervals (duration generation in TTS). From the top, the numbers of the listeners were 19, 20, and 21, respectively. (a) MSE vs. Proposed (mora). (b) MGE vs. Proposed (phoneme). (c) Proposed (mora) vs. Proposed (phoneme).

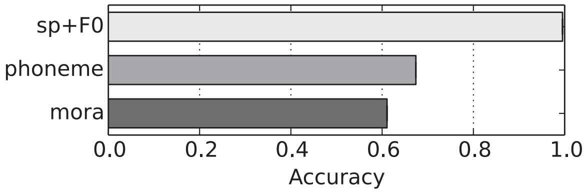

Fig. 13. Accuracy of discriminator. "sp+F0", "phoneme", and "mora" denote using the spectral parameters and F_0, phoneme durations, and mora durations for discriminating the natural and synthetic speech, respectively.

Fig. 11 shows the results. Since the score of "Proposed (sp+F0)" is much higher than those of "Proposed (sp)" and "MGE," we can confirm the effectiveness of the proposed algorithm for not only spectral parameters but also F_0.

5) Subjective Evaluation of Duration Generation: We evaluated the effect of the proposed algorithm for duration generation. We conducted a subjective evaluation using the following three methods:

[MSE:] conventional MSE
[Proposed (phoneme):] proposed algorithm applied to phoneme duration
[Proposed (mora):] proposed algorithm applied to mora duration

The preference AB test was conducted in the same manner as in the previous evaluation described in Section.

The results are shown in Fig. 12. There are no significant differences in the resulting scores. To investigate the reason, we constructed an discriminator that distinguishes conventional MSE and natural speech, and calculated the classification accuracy. We expect that our algorithm works better when the conventional generated parameters are much distinguished from the natural ones. As shown in Fig. 13, the accuracy of the discriminator that uses durations is lower than that of the discriminator that uses spectral parameters and F_0. This result infers that distribution compensation by our algorithm does not work well in

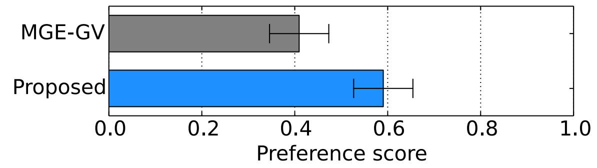

Fig. 14. Preference scores of speech quality with 95% confidence intervals (compared with the GV compensation).

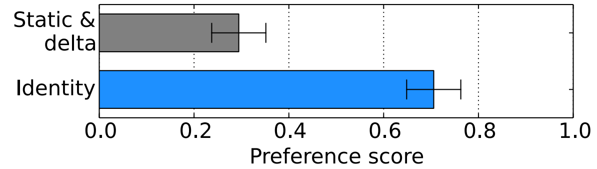

Fig. 15. Preference scores of speech quality with 95% confidence intervals (effect of the feature function which is used in anti-spoofing).

duration generation. Henceforth, we did not apply the proposed algorithm for generating durations.

6) Comparison to GV Compensation: Fig. 6 demonstrated that our method compensates the GV of the generated speech parameters. In addition, we investigate whether or not our method improves speech quality more than explicit GV compensation. We applied the post-filtering process [51] to the spectral and F_0 parameters generated by the MGE training. A preference AB test with 29 listeners was conducted by using our crowd-sourced subjective evaluation systems.

Fig. 14 shows the results. Since the score of "Proposed" is higher than that of the conventional GV post-filter ("MGE-GV"), we can conclude that our method produces more gain in speech quality than the conventional GV compensation.

7) Effect of Feature Function: We investigate whether the feature function used in anti-spoofing is effective to our method. We adopted the following two functions:

[Identity:] $\boldsymbol{\phi}(\boldsymbol{y}) = \boldsymbol{y}$
[Static & delta [52]:] $\boldsymbol{\phi}(\boldsymbol{y}) = \boldsymbol{W}\boldsymbol{y}$

"Identity" is equivalent to not using the feature function. When "Static & delta" is adopted, joint vectors of the static, delta, and delta-delta mel-cepstral coefficients and continuous F_0 are input to the discriminator. A preference AB test with 31 listeners was conducted by using our crowd-sourced subjective evaluation systems.

Fig. 15 shows the results.

Clearly, the score of "Static & delta" is much lower than that of "Identity." From this result, although "Static & delta" effectively distinguishes natural and synthetic speech, it does not improve speech quality.

8) Subjective Evaluation Using Complicated Model Architecture: Only simple Feed-Forward networks were used in the above-described evaluations. Accordingly, we confirm whether our method can improve speech quality even when more complicated networks are used. We used two-layer unidirectional LSTMs [31] as both acoustic models and discriminator. The numbers of memory cells in the acoustic models and discriminator were 256 and 128, respectively. Our method was applied to spectral and F_0 parameters. MGE ("MGE") and the proposed ("Proposed") training algorithm were compared. A preference AB test with 19 listeners was conducted by using our crowd-sourced subjective evaluation systems.

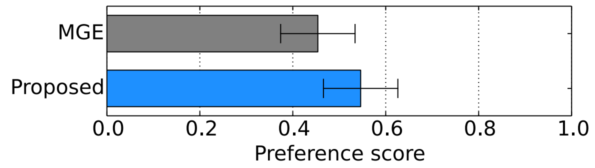

Fig. 16. Preference scores of speech quality with 95% confidence intervals (comparison in using LSTM).

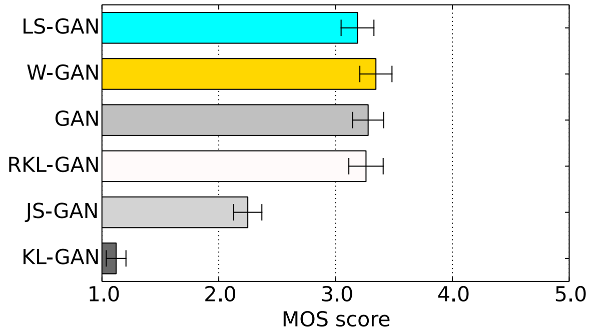

Fig. 17. MOS scores of speech quality with 95% confidence intervals (comparison in divergences of GANs).

Fig. 16 shows the results.

Since the score of "Proposed" is higher than that of "MGE," we can demonstrate that our method works for not only simple architectures but also complicated ones.

9) Effect of Divergence of GAN: As the final investigation regarding TTS, we compared speech qualities of various GANs. We adopted the following GANs:

[GAN:] (4) and (5)
[KL-GAN:] (9) and (10)
[RKL-GAN:] (12) and (13)
[JS-GAN:] (15) and (16)
[W-GAN:] (18) and (19)
[LS-GAN:] (20) and (21)

We conducted a MOS test on speech quality. The synthetic speech generated by using each GAN was presented to listeners in random order. 55 listeners participated in the assessment by using our crowdsourced subjective evaluation systems.

Fig. 17 shows the results. We can see that our method works in the case of all divergences except "KL-GAN" and "JS-GAN." Two points are noteworthy: 1) minimizing KL-divergence (KL-GAN) did not improve synthetic speech quality, but the reversed version (RKL-GAN) worked, and 2) JS-divergence did not work well, but the approximated version (GAN) worked. The best GAN in terms of synthetic speech quality was the W-GAN, whose MOS score was significantly higher than those of the LS-GAN, JS-GAN, and KL-GAN.

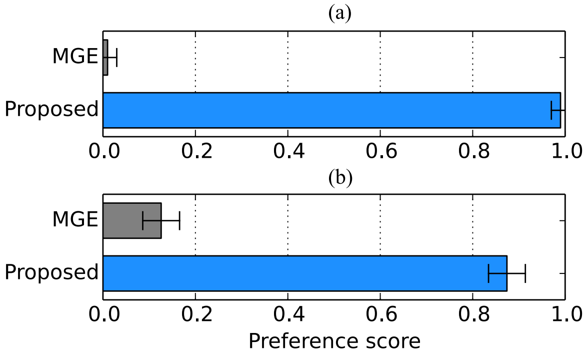

Fig. 18. Preference scores of speech quality with 95% confidence intervals (DNN-based VC). (a) Male-to-male. (b) Male-to-female.

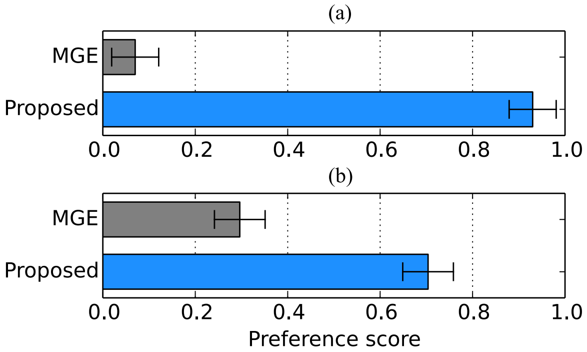

Fig. 19. Preference scores of speaker individuality with 95% confidence intervals (DNN-based VC). (a) Male-to-male. (b) Male-to-female.

C. Experimental Conditions in VC Evaluation

The experimental conditions such as dataset used in the evaluation, speech parameters, pre-processing of data, and training procedure were the same as the previous evaluations except for the dimensionality of spectral parameters and DNN architectures. We constructed DNNs for male-to-male conversion and male-to-female conversion. The hidden layers of the acoustic models and discriminator had 3 × 512 units and 3 × 256 units, respectively. The 1st-through-59th mel-cepstral coefficients were converted. The input 0th mel-cepstral coefficients were directly used as those of the converted speech. F_0 was linearly transformed, and band-aperiodicity was not transformed. Dynamic time warping was used to align total frame lengths of the input and output speech parameters.

We generated speech samples with the conventional MGE training and the proposed training algorithm. We conducted a preference AB test to evaluate the converted speech quality. We presented every pair of converted speech of the two sets in random order and had listeners select the speech sample that sounded better in quality. Similarly, an XAB test on the speaker individuality was conducted using the natural speech as a reference "X." Eight listeners participated in assessment of male-to-male conversion case, and 27 listeners participated in assessment of male-to-female conversion case using our crowdsourced subjective evaluation systems.

D. Subjective Evaluation in VC

The results of the preference tests on speech quality and speaker individuality are shown in Figs. 18 and 19, respectively. We can find that our algorithm achieves better scores in speech quality the same as the TTS evaluations. Moreover, we can see that the proposed algorithm also improves speaker individuality. We expect that the improvements are caused by compensating GVs of the generated speech parameters which affect speaker individuality [6]. These improvements were observed not only in the inter-gender case but also cross-gender case. Therefore, we have also demonstrated the effectiveness of the algorithm in DNN-based VC.

V. Conclusion

In this paper, we proposed a novel training algorithm for deep neural network (DNN)-based high-quality statistical parametric speech synthesis. The algorithm incorporates a framework of generative adversarial networks (GANs), which adversarily train generator networks and discriminator networks. In the case of proposed algorithm, acoustic models of speech synthesis are trained to deceive the discriminator that distinguishes natural and synthetic speech. Since the GAN framework minimizes the difference in distributions of natural and generated data, the acoustic models are trained to not only minimize the generation loss but also make the parameter distribution of the generated speech parameters close to that of natural speech. This is a pioneering method of GAN-based speech synthesis and can be applied not only statistical parametric approaches but also the ones such as glottal waveform synthesis [53]. We found that our algorithm compensated not only global variance but also correlation among generated speech parameters. Experimental evaluations were conducted in both DNN-based text-to-speech (TTS) synthesis and voice conversion (VC). The results demonstrate that the proposed algorithm yields significant improvements in terms of speech quality in both TTS and VC regardless of its hyper-parameter settings. We also found that the proposed algorithm incorporating the Wasserstein GAN improved synthetic speech quality the most in comparison with various GANs. In future work, we will further investigate the behavior in relation to the hyper-parameter settings, adopt feature functions which are more effective to detect synthetic speech than the identity function, and devise discriminator models with linguistic [30] dependencies.

References

[1] H. Zen, K. Tokuda, and A. Black, "Statistical parametric speech synthesis," *Speech Communi.*, vol. 51, no. 11, pp. 1039–1064, 2009.

[2] Y. Sagisaka, "Speech synthesis by rule using an optimal selection of non-uniform synthesis units," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, New York, NY, USA, Apr. 1988, pp. 679–682.

[3] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 131–142, Mar. 1988.

[4] Z.-H. Ling *et al.*, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 35–52, May 2015.

[5] K. Tokuda, Y Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proc. IEEE*, vol. 101, no. 5, pp. 1234–1252, Apr. 2013.

[6] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum likelihood estimation of spectral parameter trajectory," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007.

[7] Y. J. Wu and R. H. Wang, "Minimum generation error training for HMM-based speech synthesis," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Toulouse, France, May 2006, pp. 89–92.

[8] Z. Wu and S. King, "Improving trajectory modeling for DNN-based speech synthesis by using stacked bottleneck features and minimum trajectory error training," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 7, pp. 1255–1265, Jul. 2016.

[9] T. Toda *et al.*, "The voice conversion challenge 2016," in *Proc. INTERSPEECH*, San Francisco, CA, USA, Sep. 2016, pp. 1632–1636.

[10] Y. Ijima, T. Asami, and H. Mizuno, "Objective evaluation using association between dimensions within spectral features for statistical parametric speech synthesis," in *Proc. INTERSPEECH*, San Francisco, CA, USA, Sep. 2016, pp. 337–341.

[11] Y. Ohtani, M. Tamura, M. Morita, T. Kagoshima, and M. Akamine, "Histogram-based spectral equalization for HMM-based speech synthesis using mel-LSP," in *Proc. INTERSPEECH*, Portland, OR, USA, Sep. 2012, pp. 1155–1158.

[12] S. Takamichi, T. Toda, A. W. Black, G. Neubig, S. Sakti, and S. Nakamura, "Postfilters to modify the modulation spectrum for statistical parametric speech synthesis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 755–767, Apr. 2016.

[13] S. Takamichi, T. Toda, A. W. Black, and S. Nakamura, "Modulation spectrum-constrained trajectory training algorithm for GMM-based voice conversion," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Brisbane, QLD, Australia, Apr. 2015, pp. 4859–4863.

[14] K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Trajectory training considering global variance for speech synthesis based on neural networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 5600–5604.

[15] T. Nose and A. Ito, "Analysis of spectral enhancement using global variance in HMM-based speech synthesis," in *Proc. INTERSPEECH*, MAX Atria, Singapore, May 2014, pp. 2917–2921.

[16] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process*, Vancouver, BC, Canada, May 2013, pp. 7962–7966.

[17] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[18] Z. Wu *et al.*, "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 4, pp. 768–783, Apr. 2016.

[19] N. Chen, Y. Qian, H. Dinkel, B. Chen, and K. Yu, "Robust deep feature for spoofing detection the SJTU system for ASVspoof 2015 Challenge," in *Proc. INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2097–2101.

[20] K. Yu and S. Young, "Continuous F0 modeling for HMM based statistical parametric speech synthesis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 5, pp. 1071–1079, Jul. 2011.

[21] E. Grabe and E. L. Low, "Durational variability in speech and the rhythm class hypothesis," in *Papers in Laboratory Phonology 7*. Berlin, Germany: Mouton de Gruyter, 2002, pp. 515–546.

[22] S. Nowozin, B. Cseke, and R. Tomioka, "f-GAN: Training generative neural samplers using variational divergence minimization," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Dec. 2016, pp. 271–279.

[23] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," in *Proc. 34th Int. Conf. Mach. Learn.*, PMLR 70, 2017, pp. 214–223.

[24] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," *IEEE Int. Conf. Comput. Vision (ICCV)*, 2017, pp. 2794–2802.

[25] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2000, pp. 556–562.

[26] R. Kompass, "A generalized divergence measure for nonnegative matrix factorization," *Neural Comput.*, vol. 19, no. 3, pp. 780–891, Mar. 2007.

[27] I. Csiszár and P. C. Shields, "Information theory and statistics: A tutorial," *Found. Trends Commun. Inf. Theory*, vol. 1, no. 4, pp. 417–518, 2004.

[28] Cédric Vilani, *Optimal Transport: Old and New*. New York, NY, USA: Springer-Verlag, 2009.

[29] B. Huang, D. Ke, H. Zheng, B. Xu, Y. Xu, and K. Su, "Multi-task learning deep neural networks for speech feature denoising," in *Proc. INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2464–2468.

[30] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text-to-image synthesis," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1060–1069.

[31] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Brisbane, QLD, Australia, Apr. 2015, pp. 4470–4474.

[32] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[33] M. Tulio Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you?': Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, Aug. 2016, pp. 1135–1164.

[34] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 1718–1727.

[35] I. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks," arXiv:1701.00160, 2017.

[36] D. N. Reshef *et al.*, "Detecting novel associations in large data sets," *Science*, vol. 334, no. 6062, pp. 1518–1524, 2011.

[37] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, Lauderdale, FL, USA, Apr. 2011, pp. 315–323.

[38] K. Tanaka, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, "A hybrid approach to electrolaryngeal speech enhancement based on noise reduction and statistical excitation generation," *IEICE Trans. Inf. Syst.*, vol. E97-D, no. 6, pp. 1429–1437, Jun. 2014.

[39] S. Kang and H. Meng, "Statistical parametric speech synthesis using weighted multi-distribution deep belief network," in *Proc. INTERSPEECH*, Max Atria, Singapore, Sep. 2014, pp. 1959–1963.

[40] H. Zhang *et al.*, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," *IEEE Int. Conf. Comput. Vision (ICCV)*, 2017, pp. 5907–5915.

[41] X. Yin *et al.*, "Modeling f0 trajectories in hierarchically structured deep neural networks," *Speech Commun.*, vol. 76, pp. 82–92, 2016.

[42] T. Kaneko, H. Kameoka, N. Hojo, Y. Ijima, K. Hiramatsu, and K. Kashino, "Generative adversarial network-based postfilter for statistical parametric speech synthesis," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, New Orleans, LA, USA, Mar. 2017, pp. 4910–4914.

[43] H. Zen, Y. Agiomyrgiannakis, N. Egberts, F. Henderson, and P. Szczepaniak, "Fast, compact, and high quality LSTM-RNN based statistical parametric speech synthesizer for mobile devices," in *Proc. INTERSPEECH*, San Francisco, CA, USA, Sep. 2016, pp. 2273–2277.

[44] K. Tokuda and H. Zen, "Directly modeling voiced and unvoiced components in speech waveforms by neural networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 5640–5644.

[45] A. Oord *et al.*, "WaveNet: A generative model for raw audio," arXiv:1609.03499, 2016.

[46] Y. Sagisaka, K. Takeda, M. Abe, S. Katagiri, T. Umeda, and H. Kawahara, "A large-scale Japanese speech database," in *Proc. Int. Conf. Spoken Lang. Process.*, Kobe, Japan, Nov. 1990, pp. 1089–1092.

[47] H. Kawahara, Jo Estill, and O. Fujimura, "Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and synthesis system STRAIGHT," in *Proc. Int. Workshop Models Anal. Vocal Emissions Biomed. Appl.*, Firentze, Italy, Sep. 2001, pp. 1–6.

[48] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Maximum likelihood voice conversion based on GMM with STRAIGHT mixed excitation," in *Proc. INTERSPEECH*, Pittsburgh, PA, USA, Sep. 2006, pp. 2266–2269.

[49] H. Kawahara, I. Masuda-Katsuse, and A. D. Cheveigne, "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.*, vol. 27, no. 3/4, pp. 187–207, Apr. 1999.

[50] S. Takamichi, K. Kobayashi, K. Tanaka, T. Toda, and S. Nakamura, "The NAIST text-to-speech system for the Blizzard Challenge 2015," in *Proc. Blizzard Challenge Workshop*, Berlin, Germany, Sep. 2015.

[51] T. Toda, T. Muramatsu, and H. Banno, "Implementation of computationally efficient real-time voice conversion," in *Proc. INTERSPEECH*, Portland, OR, USA, Sep. 2012, pp. 94–97.

[52] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Proc. INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2087–2091.

[53] B. Bollepalli, L. Juvela, and P. Alku, "Generative adversarial network-based glottal waveform model for statistical parametric speech synthesis," in *Proc. INTERSPEECH*, Stockholm, Sweden, Aug. 2017, pp. 3394–3398.

Yuki Saito received the B.E. degree in engineering from the National Institution for Academic Degrees and Quality Enhancement of Higher Education, Tokyo, Japan, in 2016. He is currently working toward the M.E. degree in Information Science and Technology, Graduate School of Information Science and Technology, The University of Tokyo, Tokyo, Japan. His research interests include statistical parametric speech synthesis, machine learning, and machine intelligence. He received the 14th Best Student Presentation Award of ASJ and the 2017 IEICE ISS Student Poster Award. He is a Student Member of the Acoustical Society of Japan.

Shinnosuke Takamichi received the B.E. degree from Nagaoka University of Technology, Nagaoka, Japan, in 2011, and the M.E. and Ph.D. degrees from the Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Japan, in 2013 and 2016, respectively. He was a short-term Researcher at NICT, Kyoto, Japan, in 2013, a Visiting Researcher at Carnegie Mellon University in the United States from 2014 to 2015, and a Research Fellow (DC2) of the Japan Society for the Promotion of Science from 2014 to 2016. He is currently a Project Research Associate at The University of Tokyo. He has received more than ten paper/achievement awards including the 8th Outstanding Student Paper Award from the IEEE Japan Chapter SPS and the Itakura Prize Innovative Young Researcher Award. He is a member of ASJ, IPSJ, ISCA, and IEEE SPS.

Hiroshi Saruwatari received the B.E., M.E., and Ph.D. degrees from Nagoya University, Nagoya, Japan, in 1991, 1993, and 2000, respectively. He joined the Intelligent System Laboratory, SECOM, Co., Ltd., Tokyo, Japan, in 1993, where he was involved in research on the ultrasonic array system for acoustic imaging. He is currently a Professor in the Graduate School of Information Science and Technology, The University of Tokyo, Tokyo. His research interests include noise reduction, array signal processing, blind source separation, and sound field reproduction. He received paper awards from the IEICE in 2001 and 2006, from the Telecommunications Advancement Foundation in 2004 and 2009, and at the IEEE-IROS2005 in 2006. He received the First Prize at the IEEE MLSP2007 Data Analysis Competition for BSS. He is a Member of the IEICE, Japan VR Society, and the ASJ.