6

Modeling of Speech Parameter Sequence Considering Global Variance for HMM-Based Speech Synthesis

Tomoki Toda
Nara Institute of Science and Technology
Japan

1. Introduction

Speech technologies such as speech recognition and speech synthesis have many potential applications since speech is the main way in which most people communicate. Various linguistic sounds are produced by controlling the configuration of oral cavities to convey a message in speech communication. The produced speech sounds temporally vary and are significantly affected by coarticulation effects. Thus, it is not straightforward to segment speech signals into corresponding linguistic symbols. Moreover, the acoustics of speech vary even if the same words are uttered by the same speaker due to differences in the manner of speaking and articulatory organs. Therefore, it is essential to stochastically model them in speech processing.

The hidden Markov model (HMM) is an effective framework for modeling the acoustics of speech. Its introduction has enabled significant progress in speech and language technologies. In particular, there have been numerous efforts to develop HMM-based acoustic modeling techniques in speech recognition, and continuous density HMMs have been widely used in modern continuous speech recognition systems (Gales & Young (2008)). Moreover, several approaches have been proposed for applying the HMM-based acoustic modeling techniques to speech synthesis technologies (Donovan & Woodland (1995); Huang et al. (1996)) such as Text-to-Speech (TTS), which is ... from a given text. Recently, HMM-based speech synthesis has been proposed (Yoshimura et al. (1999)) and has generated interest owing to its various attractive features such as completely data-driven voice building, flexible voice quality control, speaker adaptation, small footprint, and so forth (Zen et al. (2009)).

A basic framework of HMM-based speech synthesis consists of training and synthesis processes. In the training process, speech parameters such as spectral envelope and fundamental frequency (F_0) are extracted from speech waveforms and then their time sequences are modeled by context-dependent phoneme HMMs. To model the dynamic characteristics of speech acoustics with HMMs, which assume piecewise constant statistics within an HMM state and conditional independence, a joint vector of static and dynamic features is usually used as an observation vector. In the synthesis process, a smoothly varying speech parameter trajectory is generated by maximizing the likelihood of a composite sentence HMM subject to a constraint between static and dynamic features with respect to not the observation vector sequence including both static and dynamic features but the static feature vector sequence (Tokuda et al. (2000)). Finally, a vocoding technique is employed

to generate a speech waveform from the generated speech parameters. This framework for directly generating speech parameters from the HMMs has the potential to be used for developing very flexible TTS systems. On the other hand, the quality of the synthetic speech is noticeably degraded compared with the original spoken audio. It is known that the static feature vectors generated from the HMMs are often oversmoothed, which is one of the main factors causing the muffled effect in synthetic speech.

There have been some attempts to model new features of speech acoustics to ensure that the generated parameters exhibit similar properties to those of natural speech, thus reducing the oversmoothing effect. As one of the most effective features for capturing properties not well modeled by traditional HMMs, Toda and Tokuda (Toda & Tokuda (2007)) proposed the global variance (GV), which is the variance of the static feature vectors calculated over a time sequence (*e.g.*, over an utterance). It has been found that the GV is inversely correlated with the oversmoothing effect. Therefore, a metric on the GV of the generated parameters effectively acts as a penalty term in the parameter generation process. Several papers (Toda & Tokuda (2007); Zen et al. (2007a; 2008)) have reported that the naturalness of synthetic speech is significantly improved by considering the GV in HMM-based speech synthesis. Moreover, it has been reported that the use of the GV is also effective for improving other statistical parametric speech synthesis techniques such as voice conversion (Toda et al. (2007)). It is not an exaggeration to say that GV modeling has significantly contributed to the recent improvements in those techniques.

This chapter presents an overview of the techniques for modeling a speech parameter sequence considering the GV for HMM-based speech synthesis. First, the traditional framework for HMM-based speech synthesis and its weaknesses are described. Then, the parameter generation algorithm considering the GV (Toda & Tokuda (2007)) is presented as an effective approach to addressing the oversmoothing problem caused by the traditional modeling process. This algorithm is capable of generating the parameter trajectory, yielding a significant improvement in synthetic speech quality while maintaining its GV close to its natural value. Furthermore, a training method considering the GV (Toda & Young (2009)) is presented, which is derived by introducing the GV-based parameter generation into the HMM training process. It is shown that this method yields some additional advantages such as the use of a consistent optimization criterion between the training and synthesis processes and the use of a closed-form solution for parameter generation, whereas only an iterative solution is available if the GV is considered in only the parameter generation process. Finally, several experimental results are presented to demonstrate that these methods yield a significant improvement in the naturalness of synthetic speech.

2. Basic framework of HMM-based speech synthesis

Figure 1 shows a schematic image of the basic training and synthesis processes of HMM-based speech synthesis. In the training process, the time sequence of the static feature vectors $\boldsymbol{c}$ is linearly transformed into a higher-dimensional space using the linear transformation function $f_o(\boldsymbol{c})$. Then, the transformed feature vector sequence consisting of static and dynamic feature vectors $\boldsymbol{o}$ is modeled with HMMs. In the synthesis process, the static feature vector sequence is determined by maximizing the likelihood of the HMMs for the static and dynamic feature vectors with respect to only the static feature vectors under the constraint given by the linear transformation function. The following subsections give more details of these training and synthesis processes.

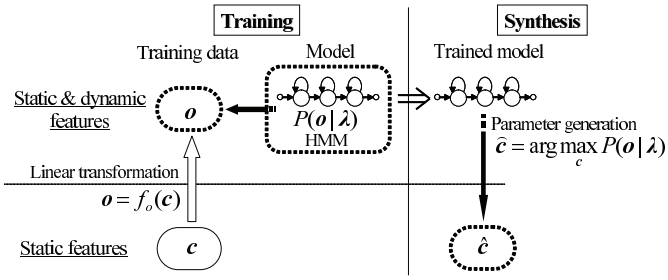


Fig. 1. Schematic image of basic training and synthesis processes of HMM-based speech synthesis.

2.1 Speech parameter sequence modeling

Let us assume a D-dimensional static feature vector of a speech parameter $\boldsymbol{c}_t = [c_t(1), c_t(2), \cdots, c_t(d), \cdots, c_t(D)]^\top$ at frame t. To suitably model the dynamic properties of the speech parameter with HMMs, the first dynamic feature vector $\Delta^{(1)}\boldsymbol{c}_t$ and the second dynamic feature vector $\Delta^{(2)}\boldsymbol{c}_t$ are calculated frame by frame as follows:

$$\Delta^{(n)}\boldsymbol{c}_t = \sum_{\tau=-L_-^{(n)}}^{L_+^{(n)}} w^{(n)}(\tau)\boldsymbol{c}_{t+\tau}, \quad n = 1, 2, \tag{1}$$

where $w^{(n)}(t)$ is a regression coefficient used to calculate the dynamic features from the current frame, $L_-^{(n)}$ previous frames, and $L_+^{(n)}$ succeeding frames,[1] and then a joint static and dynamic feature vector $\boldsymbol{o}_t = \left[\boldsymbol{c}_t^\top, \Delta^{(1)}\boldsymbol{c}_t^\top, \Delta^{(2)}\boldsymbol{c}_t^\top\right]^\top$ is used as the observation vector. The time sequence vectors of the joint static and dynamic feature vector $\boldsymbol{o}_t$ and the static feature vector $\boldsymbol{c}_t$ are written as $\boldsymbol{o} = \left[\boldsymbol{o}_1^\top, \boldsymbol{o}_2^\top, \cdots, \boldsymbol{o}_t^\top, \cdots, \boldsymbol{o}_T^\top\right]^\top$ and $\boldsymbol{c} = \left[\boldsymbol{c}_1^\top, \boldsymbol{c}_2^\top, \cdots, \boldsymbol{c}_t^\top, \cdots, \boldsymbol{c}_T^\top\right]^\top$, respectively. The relationship between these two time sequence vectors is represented as a linear transformation as follows:

$$\boldsymbol{o} = \boldsymbol{W}\boldsymbol{c}, \tag{2}$$

where $\boldsymbol{W}$ is the $3DT$-by-DT matrix written as

$$\boldsymbol{W} = [\boldsymbol{W}_1, \boldsymbol{W}_2, \cdots, \boldsymbol{W}_t, \cdots, \boldsymbol{W}_T]^\top \otimes \boldsymbol{I}_{D\times D}, \tag{3}$$

$$\boldsymbol{W}_t = \left[\boldsymbol{w}_t^{(0)}, \boldsymbol{w}_t^{(1)}, \boldsymbol{w}_t^{(2)}\right], \tag{4}$$

$$\boldsymbol{w}_t^{(n)} = \Big[\underbrace{0}_{1\text{st}}, \cdots, 0, \underbrace{w^{(n)}(-L_-^{(n)})}_{\left(t-L_-^{(n)}\right)\text{-th}}, \cdots, \underbrace{w^{(n)}(0)}_{(t)\text{-th}}, \cdots, \underbrace{w^{(n)}(L_+^{(n)})}_{\left(t+L_+^{(n)}\right)\text{-th}}, 0, \cdots, \underbrace{0}_{T\text{-th}}\Big]^\top, \tag{5}$$

[1] For example, when the number of frames to be used for calculating dynamic features is set to $L_-^{(1)} = L_+^{(1)} = 1$ and $L_-^{(2)} = L_+^{(2)} = 1$, the regression coefficients can be set to $w^{(1)}(t-1) = -0.5$, $w^{(1)}(t) = 0$, $w^{(1)}(t+1) = 0.5$, $w^{(2)}(t-1) = 1$, $w^{(2)}(t) = -2$, and $w^{(2)}(t+1) = 1$.

where $L_{-}^{(0)} = L_{+}^{(0)} = 0$, and $w^{(0)}(0) = 1$. The operator $\otimes$ denotes the Kronecker product. The matrix $\boldsymbol{I}_{D\times D}$ denotes the D-by-D identity matrix. It is important to note that the joint static and dynamic feature vector sequence $\boldsymbol{o}$ is generated from the static feature vector sequence $\boldsymbol{c}$ using the linear transformation function given by Eq. (2).

The observation sequence vector $\boldsymbol{o}$ is modeled by context-dependent phoneme HMMs, whose the parameter set is denoted by $\boldsymbol{\lambda}$. The state output probability density function (p.d.f.) of $\boldsymbol{o}_t$ at an HMM state q is usually modeled by a single multivariate Gaussian distribution as follows:

$$\mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_q, \boldsymbol{U}_q\right) = \frac{1}{\sqrt{(2\pi)^{3D}|\boldsymbol{U}_q|}} \exp\left(-\frac{1}{2}\left(\boldsymbol{o}_t - \boldsymbol{\mu}_q\right)^{\top} \boldsymbol{U}_q^{-1}\left(\boldsymbol{o}_t - \boldsymbol{\mu}_q\right)\right), \tag{6}$$

where $\boldsymbol{\mu}_q$ and $\boldsymbol{U}_q$ are the mean vector and the covariance matrix, respectively. The p.d.f. of the observation sequence vector $\boldsymbol{o}$ given an HMM state sequence $\boldsymbol{q} = \{q_1, q_2, \cdots, q_t, \cdots, q_T\}$ is written as

$$\begin{aligned} P\left(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}\right) &= \prod_{t=1}^{T} \mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_{q_t}, \boldsymbol{U}_{q_t}\right) \\ &= \mathcal{N}\left(\boldsymbol{o}; \boldsymbol{\mu}_{\boldsymbol{q}}, \boldsymbol{U}_{\boldsymbol{q}}\right), \end{aligned} \tag{7}$$

where $\boldsymbol{\mu}_{\boldsymbol{q}}$ and $\boldsymbol{U}_{\boldsymbol{q}}$ are the $3D$-by-1 mean vector and the $3D$-by-$3D$ covariance matrix, respectively, which are given by

$$\boldsymbol{\mu}_{\boldsymbol{q}} = \left[\boldsymbol{\mu}_{q_1}^{\top}, \boldsymbol{\mu}_{q_2}^{\top}, \cdots, \boldsymbol{\mu}_{q_t}^{\top}, \cdots, \boldsymbol{\mu}_{q_T}^{\top}\right]^{\top}, \tag{8}$$

$$\begin{aligned} \boldsymbol{U}_{\boldsymbol{q}} &= \mathrm{diag}\left[\boldsymbol{U}_{q_1}, \boldsymbol{U}_{q_2}, \cdots, \boldsymbol{U}_{q_t}, \cdots, \boldsymbol{U}_{q_T}\right] \\ &= \begin{bmatrix} \boldsymbol{U}_{q_1} & & & & \\ & \boldsymbol{U}_{q_2} & & & \\ & & \ddots & & \\ & & & \boldsymbol{U}_{q_T} \end{bmatrix}. \end{aligned} \tag{9}$$

The operator $\mathrm{diag}[\cdot]$ denotes the transformation from a rectangular matrix (or a vector) to a block diagonal matrix (or a diagonal matrix). The likelihood function given the HMM set for the observation sequence vector $\boldsymbol{o}$ is given by

$$\begin{aligned} P\left(\boldsymbol{o}|\boldsymbol{\lambda}\right) &= \sum_{\text{all } \boldsymbol{q}} P\left(\boldsymbol{o}, \boldsymbol{q}|\boldsymbol{\lambda}\right) \\ &= \sum_{\text{all } \boldsymbol{q}} P\left(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}\right) P\left(\boldsymbol{q}|\boldsymbol{\lambda}\right), \end{aligned} \tag{10}$$

where $P\left(\boldsymbol{q}|\boldsymbol{\lambda}\right)$ is usually modeled by state transition probabilities in speech recognition. However, in speech synthesis explicit duration models are often incorporated into the HMMs to model the temporal structure of a speech parameter sequence appropriately (Yoshimura et al. (1999); Zen et al. (2007b)). In training, the HMM parameter set $\boldsymbol{\lambda}$ is optimized for the given observation sequence vectors $\boldsymbol{o}^{(1)}, \boldsymbol{o}^{(2)}, \cdots, \boldsymbol{o}^{(k)}, \cdots, \boldsymbol{o}^{(K)}$ in the sense of maximum likelihood

(ML) as follows:

$$\hat{\lambda} = \arg\max_{\lambda} \prod_{k=1}^{K} P\left(o^{(k)}|\lambda\right), \tag{11}$$

where K is the total number of observation sequence vectors.

2.2 Parameter generation based on maximum likelihood criterion

In synthesis, a time sequence of static feature vectors is determined by maximizing the HMM likelihood under the condition given by Eq. (2) as follows:

$$\hat{c} = \arg\max_{c} P\left(o|\lambda\right) \qquad \text{subject to} \quad o = Wc. \tag{12}$$

To reduce the computational cost, the HMM likelihood is usually approximated with a single HMM state sequence as follows:

$$P(o|\lambda) \simeq P(o|q,\lambda)P(q|\lambda), \tag{13}$$

and then the HMM state sequence and the static feature vector sequence are sequentially determined. First a suboptimum HMM state sequence is determined by

$$\hat{q} = \arg\max P(q|\lambda). \tag{14}$$

Then, the static feature vector sequence is determined by maximizing the HMM likelihood given the HMM state sequence q as follows:

$$\hat{c} = \arg\max P(o|q,\lambda) \qquad \text{subject to} \quad o = Wc. \tag{15}$$

The objective function $\mathcal{L}_q$ to be maximized with respect to the static feature vector sequence is given by

$$\begin{aligned} \mathcal{L}_q &= \log P(o|q,\lambda) && (16)\\ &\propto -\frac{1}{2}o^\top U_q^{-1}o + o^\top U_q^{-1}\mu_q \\ &= -\frac{1}{2}c^\top W^\top U_q^{-1}Wc + c^\top W^\top U_q^{-1}\mu_q \\ &= -\frac{1}{2}c^\top R_q c + c^\top r_q, && (17) \end{aligned}$$

where

$$R_q = W^\top U_q^{-1} W, \tag{18}$$

$$r_q = W^\top U_q^{-1}\mu_q. \tag{19}$$

The ML estimate of the static feature vector sequence $\overline{c}_q$ is given by

$$\overline{c}_q = P_q r_q, \tag{20}$$

$$P_q = R_q^{-1}. \tag{21}$$

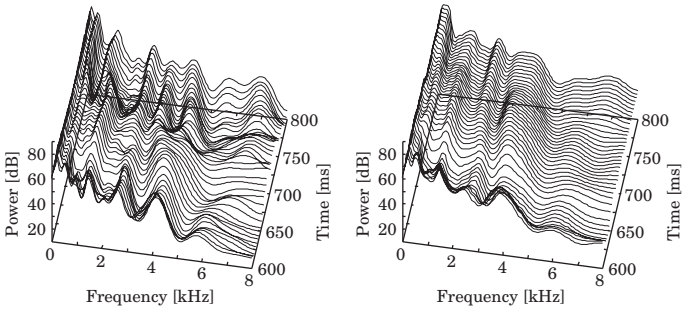


Fig. 2. An example of natural and generated spectral segments.

Since the matrix $\boldsymbol{P}_{\boldsymbol{q}}$ is generally full owing to the inverse of the band matrix $\boldsymbol{R}_{\boldsymbol{q}}$, the state output p.d.f. at each HMM state affects the ML estimates of the static feature vectors at all frames over a time sequence. This parameter generation algorithm is capable of generating speech parameter trajectories that vary frame by frame from the p.d.f. sequence corresponding to discrete state sequences so that the generated trajectories exhibit suitable static and dynamic properties.

The calculation of the ML estimate is efficiently performed by the Cholesky decomposition of $\boldsymbol{R}_{\boldsymbol{q}}$ and the forward/backward substitution operation. It is also possible to implement a recursive estimation process so as to generate the static feature vectors frame by frame (Tokuda et al. (1995)). Furthermore, although only the case of using a single multivariate Gaussian distribution as the state output p.d.f. is described in this section, Gaussian mixture models can also be employed in this framework by using the Expectation-Maximization (EM) algorithm (Tokuda et al. (2000)).

2.3 Oversmoothing effect

In HMM-based speech synthesis, speech samples synthesized with the generated speech parameters often sound muffled. One of the factors causing the muffled sound is the oversmoothing of the generated speech parameters. **Figure 2** shows an example of natural and generated spectral segments. It can be observed from this figure that the generated spectra are often excessively smoothed compared with the natural spectra. The statistical modeling process with HMMs tends to remove the details of spectral structures. Although this smoothing results in reduced error in the generation of spectra, it also causes the degradation of naturalness of synthetic speech because the removed structures are still necessary for synthesizing high-quality speech.

3. Parameter generation considering global variance

To reduce oversmoothing, a parameter generation algorithm considering the GV has been proposed (Toda & Tokuda (2007)). A schematic image of the training and synthesis processes is shown in **Figure 3**. In the training process, the linearly transformed feature vector sequence consisting of static and dynamic feature vectors $\boldsymbol{o}$ is modeled with HMMs in the same manner as discussed in **Section 2**. Also, the time sequence of the static feature vectors $\boldsymbol{c}$ is nonlinearly transformed into the GV $\boldsymbol{v}$ using the nonlinear transformation function $f_v(\boldsymbol{c})$, and then its p.d.f. is modeled with a continuous density distribution. In the synthesis process, the static feature vector sequence is determined by maximizing the product of the HMM likelihood and

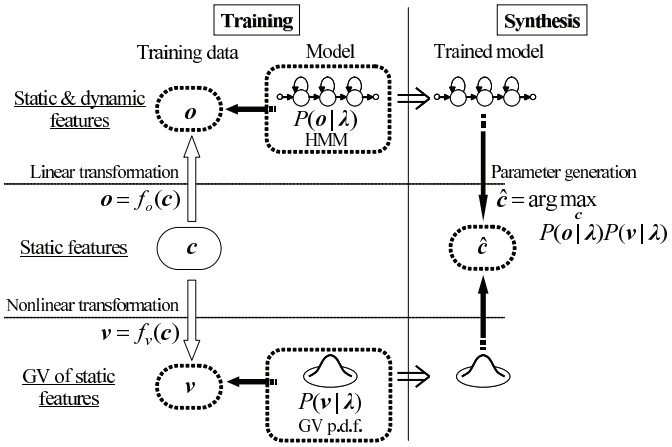


Fig. 3. Schematic image of training and synthesis processes of HMM-based speech synthesis with GV-based parameter generation algorithm.

the GV likelihood. The following subsections give more details of these training and synthesis processes.

3.1 Global Variance (GV)

The GV vector $\boldsymbol{v}(\boldsymbol{c}) = [v(1), \cdots, v(d), \cdots, v(D)]^\top$ of a static feature vector sequence $\boldsymbol{c}$ is calculated by

$$v(d) = \frac{1}{T} \sum_{t=1}^{T} \left(c_t(d) - \langle c(d) \rangle\right)^2, \tag{22}$$

where

$$\langle c(d) \rangle = \frac{1}{T} \sum_{\tau=1}^{T} c_\tau(d). \tag{23}$$

The GV is often calculated utterance by utterance.

Figure 4 shows a time sequence of the 2nd mel-cepstral coefficients extracted from natural speech and that generated from the HMM, where mel-cepstrum is one of the most effective spectral parameters (Tokuda et al. (1994)). It can be observed that the GV of the generated mel-cepstra (given by Eq. (20)) is smaller than that of the natural ones. The ML criterion usually makes the generated trajectory close to the mean vector sequence of the HMM. Consequently, the reduction of GV is often observed.

3.2 Parameter generation algorithm considering GV

To consider the GV in the parameter generation process, the p.d.f. of the GV is modeled by a single multivariate Gaussian distribution, which is given by

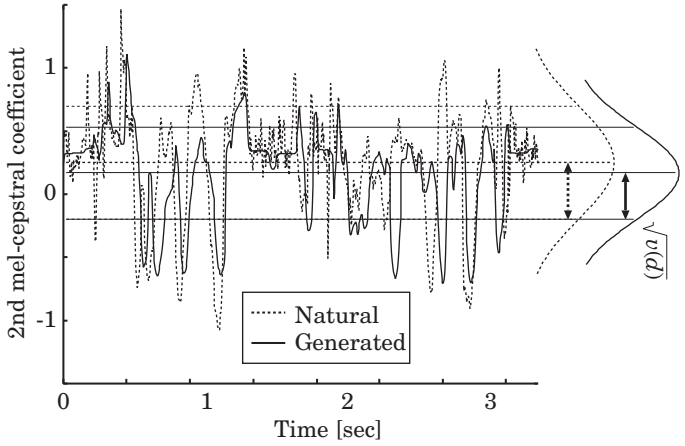


Fig. 4. Natural and generated mel-cepstrum sequences. The square root of the GV of each sequence is shown as a bidirectional arrow.

$$P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}_v) = \mathcal{N}(\boldsymbol{v}(\boldsymbol{c}); \boldsymbol{\mu}_v, \boldsymbol{U}_v) = \frac{1}{\sqrt{(2\pi)^D |\boldsymbol{U}_v|}} \exp\left(-\frac{1}{2}(\boldsymbol{v}(\boldsymbol{c}) - \boldsymbol{\mu}_v)^\top \boldsymbol{U}_v^{-1}(\boldsymbol{v}(\boldsymbol{c}) - \boldsymbol{\mu}_v)\right), \tag{24}$$

where $\boldsymbol{\lambda}_v$ denotes the GV parameter set consisting of the mean vector $\boldsymbol{\mu}_v$ and the covariance matrix $\boldsymbol{U}_v$. The GV parameter set $\boldsymbol{\lambda}_v$ and the HMM parameter set $\boldsymbol{\lambda}$ are independently trained from speech samples in the training data.

In the synthesis process, given a suboptimum HMM state sequence $\boldsymbol{q}$, the static feature vector sequence is determined by maximizing a new likelihood function given by a product of the HMM likelihood for the static and dynamic feature vectors and the GV likelihood as follows:

$$\hat{\boldsymbol{c}} = \operatorname{argmax} P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}) P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}_v)^{\omega T} \qquad \text{subject to} \quad \boldsymbol{o} = \boldsymbol{W}\boldsymbol{c}, \tag{25}$$

where the constant ω denotes the GV weight, used for controlling the balance between the two likelihoods, which is usually set to the ratio between the number of dimensions of vectors $\boldsymbol{v}(\boldsymbol{c})$ and $\boldsymbol{o}$ (*i.e.*, $\omega = 3$). Note that this likelihood function with an additional constraint on the GV of the generated trajectory is still a function of the static feature vector sequence $\boldsymbol{c}$. The GV likelihood $P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}_v)$ can be viewed as a penalty term for a reduction of the GV. The objective function $\mathcal{L}_{\boldsymbol{q}}^{(GV)}$ to be maximized with respect to the static feature vector sequence is written as

$$\mathcal{L}_{\boldsymbol{q}}^{(GV)} = \log P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}) + \omega T \log P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}_v) \tag{26}$$

$$\propto -\frac{1}{2}\boldsymbol{c}^\top \boldsymbol{R}_{\boldsymbol{q}} \boldsymbol{c} + \boldsymbol{c}^\top \boldsymbol{r}_{\boldsymbol{q}} + \omega T \left(-\frac{1}{2}\boldsymbol{v}(\boldsymbol{c})^\top \boldsymbol{U}_v^{-1} \boldsymbol{v}(\boldsymbol{c}) + \boldsymbol{v}(\boldsymbol{c})^\top \boldsymbol{U}_v^{-1} \boldsymbol{\mu}_v\right). \tag{27}$$

This objective function is equivalent to the traditional function given by Eq. (16) when the GV weight ω is set to 0. To determine the static feature vector sequence maximizing the objective function, an iterative process for updating $\hat{\boldsymbol{c}}$ employing the gradient method is necessary as

follows:

$$\hat{\boldsymbol{c}}^{(i+1)\text{-th}} = \hat{\boldsymbol{c}}^{(i)\text{-th}} + \alpha \cdot \delta\hat{\boldsymbol{c}}^{(i)\text{-th}}, \tag{28}$$

where α is the step size parameter. The following two gradient methods are basically employed to calculate the vector $\delta\hat{\boldsymbol{c}}^{(i)\text{-th}}$.

Steepest descent algorithm: Using the steepest descent algorithm, $\delta\hat{\boldsymbol{c}}^{(i)\text{-th}}$ is written as

$$\delta\hat{\boldsymbol{c}}^{(i)\text{-th}} = \left.\frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(GV)}}{\partial \boldsymbol{c}}\right|_{\boldsymbol{c}=\hat{\boldsymbol{c}}^{(i)\text{-th}}}. \tag{29}$$

The first derivative is calculated by

$$\frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(GV)}}{\partial \boldsymbol{c}} = -\boldsymbol{R}_{\boldsymbol{q}}\boldsymbol{c} + \boldsymbol{r}_{\boldsymbol{q}} + \omega \boldsymbol{x}, \tag{30}$$

$$\boldsymbol{x} = \left[\boldsymbol{x}_1^\top, \boldsymbol{x}_2^\top, \cdots, \boldsymbol{x}_t^\top, \cdots, \boldsymbol{x}_T^\top\right]^\top, \tag{31}$$

$$\boldsymbol{x}_t = \left[x_t(1), x_t(2), \cdots, x_t(d), \cdots, x_t(D)\right]^\top, \tag{32}$$

$$x_t(d) = -2\left(c_t(d) - \langle c(d)\rangle\right)\left(\boldsymbol{v}(\boldsymbol{c}) - \boldsymbol{\mu}_v\right)^\top \boldsymbol{p}_v^{(d)}, \tag{33}$$

where $\boldsymbol{p}_v^{(d)}$ is the d-th column vector of the precision matrix $\boldsymbol{P}_v (= \boldsymbol{U}_v^{-1})$.

Newton-Raphson method: If the initial value of the static feature vector sequence $\hat{\boldsymbol{c}}^{(0)\text{-th}}$ is close to the optimum value, the Newton-Raphson method using not only the first derivative but also the second derivative, *i.e.*, the Hessian matrix, may also be used. The vector $\delta\hat{\boldsymbol{c}}^{(i)\text{-th}}$ is written as

$$\delta\hat{\boldsymbol{c}}^{(i)\text{-th}} = -\left(\frac{\partial^2 \mathcal{L}_{\boldsymbol{q}}^{(GV)}}{\partial \boldsymbol{c}\partial \boldsymbol{c}^\top}\right)^{-1} \left.\frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(GV)}}{\partial \boldsymbol{c}}\right|_{\boldsymbol{c}=\hat{\boldsymbol{c}}^{(i)\text{-th}}}. \tag{34}$$

Because a Hessian matrix is not always a positive-definite matrix, the following second derivative, approximated using only diagonal elements, is used:

$$\frac{\partial^2 \mathcal{L}_{\boldsymbol{q}}^{(GV)}}{\partial \boldsymbol{c}\partial \boldsymbol{c}^\top} \simeq \operatorname{diag}\left[-\operatorname{diag}^{-1}\left[\boldsymbol{R}_{\boldsymbol{q}}\right] + \omega \boldsymbol{y}\right], \tag{35}$$

$$\boldsymbol{y} = \left[\boldsymbol{y}_1^\top, \boldsymbol{y}_2^\top, \cdots, \boldsymbol{y}_t^\top, \cdots, \boldsymbol{y}_T^\top\right]^\top, \tag{36}$$

$$\boldsymbol{y}_t = \left[\frac{\delta x_t(1)}{\delta c_t(1)}, \frac{\delta x_t(2)}{\delta c_t(2)}, \cdots, \frac{\delta x_t(d)}{\delta c_t(d)}, \cdots, \frac{\delta x_t(D)}{\delta c_t(D)}\right]^\top, \tag{37}$$

$$\frac{\delta x_t(d)}{\delta c_t(d)} = -\frac{2}{T}\left\{(T-1)\left(\boldsymbol{v}(\boldsymbol{c}) - \boldsymbol{\mu}_v\right)^\top \boldsymbol{p}_v^{(d)} + 2\left(c_t(d) - \langle c(d)\rangle\right)^2 p_v^{(d)}(d)\right\}, \tag{38}$$

where the operator $\operatorname{diag}^{-1}[\cdot]$ denotes the inverse operation of $\operatorname{diag}[\cdot]$; *i.e.*, the extraction of only the diagonal elements (or block diagonal matrices) from a square matrix.

There are two main methods for setting the initial value of the static feature vector sequence $\hat{\boldsymbol{c}}^{(0)\text{-th}}$. One is to use the ML estimate $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ given by Eq. (20) in the traditional parameter generation process. The other is to use the static feature vector sequence $\boldsymbol{c}'$ linearly converted from the ML estimate so that its GV is equivalent to the mean vector of the GV p.d.f., $\boldsymbol{\mu}_v$, as follows:

$$c'_t(d) = \sqrt{\frac{\mu_v(d)}{v(d)}}\left(c_t(d) - \langle c(d)\rangle\right) + \langle c(d)\rangle\,, \tag{39}$$

where $\mu_v(d)$ is the d-th element of the mean vector $\boldsymbol{\mu}_v$. The former sequence maximizes the HMM likelihood $P(\boldsymbol{o}|\boldsymbol{q},\boldsymbol{\lambda})$, while the latter sequence maximizes the GV likelihood $P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}_v)$. It is reasonable to start the iterative update from the static feature vector sequence yielding a higher value of the objective function given by Eq. (26).

3.3 Effectiveness of considering GV

It is well known that postfiltering to increase the sharpness of spectral peaks is effective for improving synthetic speech quality (Koishida et al. (1995)), in which it is necessary to empirically adjust a parameter to control the degree of emphasis. Moreover, this parameter is usually kept constant over different frames. On the other hand, when considering the GV, at a certain dimension the trajectory movements are greatly emphasized, but at another dimension they remain almost the same. The degree of emphasis varies between individual dimensions and frames, and it is automatically determined from the objective function given by Eq. (26). This process may be regarded as statistical postfiltering.

Using more mixture components to model the probability density also reduces oversmoothing. However, it also causes another problem of overtraining due to an increase in the number of model parameters, which often causes performance degradation for data samples not included in the training data. One of the advantages of considering the GV is that the number of parameters is kept almost equal to that when not considering the GV. In addition, since the proposed framework is based on a statistical process, it retains many advantages of statistical parametric speech synthesis, such as allowing model adaptation (Yamagishi et al. (2009)) in a manner supported mathematically.

3.4 Weakness

The main weakness of considering the GV is the inconsistency between the training and synthesis criteria. In the training, the likelihood for the joint static and dynamic feature vectors is used to optimize the HMM parameters and the likelihood for the GV is used to optimize the GV p.d.f. parameters. These two models are independently optimized. On the other hand, in the synthesis, the product of these two likelihoods is used to optimize only the static feature vectors. Consequently, the trained model parameters are not optimum for this parameter generation process.

The GV p.d.f. given by Eq. (24) is context-independent. Hence, it does not capture variations of the GV caused by different contextual factors. A context-dependent model may be used as the GV p.d.f. to capture them. However, if the GV is calculated utterance by utterance, the number of GV samples used to train the context-dependent model is relatively small; *i.e.*, only the number of utterances in the training data. Therefore, the number of context-dependent GV p.d.f.s is often limited to avoid the overtraining problem.

The parameter generation process with the GV requires the gradient method. Thus, it has a greater computational cost than the traditional parameter generation process, for which the closed-form solution can be used.

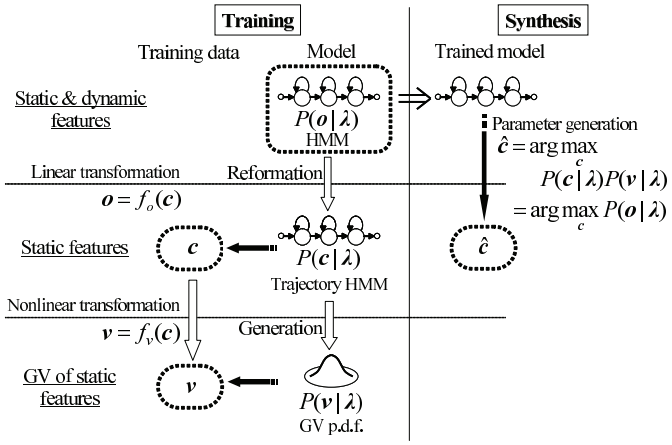


Fig. 5. Schematic image of training and synthesis processes of HMM-based speech synthesis with GV-constrained trajectory training algorithm.

4. HMM training considering global variance

The traditional framework described in **Section 2** also suffers from the inconsistency between the training and synthesis criteria. To address this issue, Zen *et al*. (Zen et al. (2007c)) proposed the trajectory HMM, which is derived by imposing an explicit relationship between static and dynamic features on the traditional HMM. This method allows the utilization of a unified criterion, *i.e.*, the trajectory likelihood, in both training and synthesis processes. In a similar spirit, Wu and Wang (Wu & Wang (2006)) proposed minimum generation error (MGE) training. This method optimizes the HMM parameters so that the error between the generated and natural parameters is minimized.

Inspired by these approaches, the idea of considering the GV has been introduced to the model training process to make it possible to use the same objective function consisting of the GV metric in both the training process and the synthesis process (Toda & Young (2009)). A schematic image of the training and synthesis processes is shown in **Figure 5**. The basic HMMs that model the p.d.f. of the joint static and dynamic feature vector sequence are reformulated as the trajectory HMM, which models the p.d.f. of the static feature vector sequence by imposing the constraint given by the linear transformation function $f_o(\boldsymbol{c})$. Furthermore, the GV p.d.f. is generated from the trajectory HMM based on the constraint given by the nonlinear transformation function $f_v(\boldsymbol{c})$. Then, the basic HMM parameters are optimized so that the objective function defined as the product of the likelihood of the trajectory HMM and the likelihood of the generated GV p.d.f. is maximized. In the synthesis process, the static feature vector sequence is determined by maximizing the same objective function. The following subsections give more details of these training and synthesis processes.

4.1 Trajectory HMM

The traditional HMM is reformulated as a trajectory HMM by imposing an explicit relationship between static and dynamic features, which is given by Eq. (2) (Zen et al. (2007c)).

The p.d.f. of $\boldsymbol{c}$ in the trajectory HMM is given by

$$P(\boldsymbol{c}|\boldsymbol{\lambda}) = \sum_{\text{all } \boldsymbol{q}} P(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda})\, P(\boldsymbol{q}|\boldsymbol{\lambda}), \tag{40}$$

$$\begin{aligned} P(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda}) &= \frac{1}{Z_{\boldsymbol{q}}} P(\boldsymbol{o}|\boldsymbol{q},\boldsymbol{\lambda}) \\ &= \mathcal{N}(\boldsymbol{c}; \overline{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}), \end{aligned} \tag{41}$$

where the normalization term $Z_{\boldsymbol{q}}$ is given by

$$\begin{aligned} Z_{\boldsymbol{q}} &= \int P(\boldsymbol{o}|\boldsymbol{q},\boldsymbol{\lambda}) d\boldsymbol{c} \\ &= \frac{\sqrt{(2\pi)^{DT}|\boldsymbol{P}_{\boldsymbol{q}}|}}{\sqrt{(2\pi)^{3DT}|\boldsymbol{U}_{\boldsymbol{q}}|}} \exp\left(-\frac{1}{2}(\boldsymbol{\mu}_{\boldsymbol{q}}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}} - \boldsymbol{r}_{\boldsymbol{q}}^{\top}\boldsymbol{P}_{\boldsymbol{q}}\boldsymbol{r}_{\boldsymbol{q}})\right). \end{aligned} \tag{42}$$

In Eq. (41), the mean vector $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ (given by Eq. (20)) varies within the states, and the interframe correlation is modeled by the temporal covariance matrix $\boldsymbol{P}_{\boldsymbol{q}}$ (given by Eq. (21)) even when using the same number of model parameters as in the traditional HMM. Note that the mean vector of the trajectory HMM is equivalent to the ML estimate of the generated static feature sequence in the traditional parameter generation process. Namely, the traditional parameter generation process is equivalent to the maximization process of the likelihood function of the trajectory HMM with respect to the static feature vector sequence. The utilization of the trajectory likelihood as a unified criterion in both training and synthesis processes makes it possible to optimize the HMM parameters for parameter generation.

4.2 GV-constrained trajectory training

The parameter generation process considering the GV given by Eq. (25) has been modified and integrated into a training framework as follows:

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} P(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda}) P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{q},\boldsymbol{\lambda})^{\omega T}, \tag{43}$$

where $P(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda})$ is given by Eq. (41) and $P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{q},\boldsymbol{\lambda})$ is the modified GV p.d.f., which is given by

$$P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{q},\boldsymbol{\lambda}) = \mathcal{N}\left(\boldsymbol{v}(\boldsymbol{c}); \boldsymbol{v}(\overline{\boldsymbol{c}}_{\boldsymbol{q}}), \boldsymbol{U}_v\right). \tag{44}$$

Note that the mean vector of the GV p.d.f. is defined as the GV of the mean vector of the trajectory HMM, which is equivalent to the GV of the generated parameters from the HMMs given by Eq. (20). Hence, the GV likelihood $P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{q},\boldsymbol{\lambda})$ acts as a penalty term to make the GV of the generated parameters close to that of the natural ones. The balance between the two likelihoods $P(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda})$ and $P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{q},\boldsymbol{\lambda})$ is controlled by the GV weight ω. The objective function $\mathcal{L}_{\boldsymbol{q}}^{(GV)'}$ used in both training and synthesis processes is given by

$$\mathcal{L}_{\boldsymbol{q}}^{(GV)'} = \log P(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda}) + \omega T \log P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{q},\boldsymbol{\lambda}). \tag{45}$$

Given the HMM state sequence $\boldsymbol{q}$,[2] the GV weight ω, and the GV covariance matrix $\boldsymbol{U}_v$, the HMM parameter set is optimized by maximizing the proposed objective function $\mathcal{L}_{\boldsymbol{q}}^{(GV)'}$. The mean vectors and diagonal precision matrices at all HMM states (from 1 to N), which are given by

$$\boldsymbol{m} = \left[\boldsymbol{\mu}_1^\top, \boldsymbol{\mu}_2^\top, \cdots, \boldsymbol{\mu}_N^\top\right]^\top, \tag{46}$$

$$\boldsymbol{\Sigma}^{-1} = \left[\boldsymbol{U}_1^{-1}, \boldsymbol{U}_2^{-1}, \cdots, \boldsymbol{U}_N^{-1}\right]^\top, \tag{47}$$

are simultaneously updated since they depend on each other. The mean vectors $\boldsymbol{m}$ are iteratively updated using the following gradient:

$$\frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(GV)'}}{\partial \boldsymbol{m}} = \boldsymbol{A}_{\boldsymbol{q}}^\top \boldsymbol{U}_{\boldsymbol{q}}^{-1} \boldsymbol{W} \left(\boldsymbol{c} - \overline{\boldsymbol{c}}_{\boldsymbol{q}} + \omega \boldsymbol{P}_{\boldsymbol{q}} \overline{\boldsymbol{x}}_{\boldsymbol{q}}\right), \tag{48}$$

where

$$\overline{\boldsymbol{x}}_{\boldsymbol{q}} = \left[\overline{\boldsymbol{x}}_{\boldsymbol{q},1}^\top, \overline{\boldsymbol{x}}_{\boldsymbol{q},2}^\top, \cdots, \overline{\boldsymbol{x}}_{\boldsymbol{q},t}^\top, \cdots, \overline{\boldsymbol{x}}_{\boldsymbol{q},T}^\top\right]^\top, \tag{49}$$

$$\overline{\boldsymbol{x}}_{\boldsymbol{q},t} = \left[\overline{x}_{\boldsymbol{q},t}(1), \overline{x}_{\boldsymbol{q},t}(2), \cdots, \overline{x}_{\boldsymbol{q},t}(d), \cdots, \overline{x}_{\boldsymbol{q},t}(D)\right]^\top, \tag{50}$$

$$\overline{x}_{\boldsymbol{q},t}(d) = -2\left(\overline{c}_{\boldsymbol{q},t}(d) - \left\langle \overline{c}_{\boldsymbol{q}}(d) \right\rangle\right)\left(\boldsymbol{v}(\overline{\boldsymbol{c}}_{\boldsymbol{q}}) - \boldsymbol{v}(\boldsymbol{c})\right)^\top \boldsymbol{p}_v^{(d)}, \tag{51}$$

and $\boldsymbol{A}_{\boldsymbol{q}}$ is a $3DT$-by-$3DN$ matrix whose elements are 0 or 1 depending on the state sequence $\boldsymbol{q}$. The precision matrices $\boldsymbol{\Sigma}^{-1}$ are iteratively updated using the following gradient:

$$\begin{aligned}\frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(GV)'}}{\partial \boldsymbol{\Sigma}^{-1}} = \frac{1}{2}\boldsymbol{A}_{\boldsymbol{q}}^\top \mathrm{diag}^{-1}\Big[&\boldsymbol{W}(\boldsymbol{P}_{\boldsymbol{q}} + \overline{\boldsymbol{c}}_{\boldsymbol{q}}\overline{\boldsymbol{c}}_{\boldsymbol{q}}^\top - \boldsymbol{c}\boldsymbol{c}^\top)\boldsymbol{W}^\top - 2\boldsymbol{\mu}_{\boldsymbol{q}}(\overline{\boldsymbol{c}}_{\boldsymbol{q}} - \boldsymbol{c}_{\boldsymbol{q}})^\top \boldsymbol{W}^\top \\ &+2\omega \boldsymbol{W}\boldsymbol{P}_{\boldsymbol{q}}\overline{\boldsymbol{x}}_{\boldsymbol{q}}(\boldsymbol{\mu}_{\boldsymbol{q}} - \boldsymbol{W}\overline{\boldsymbol{c}}_{\boldsymbol{q}})^\top\Big].\end{aligned} \tag{52}$$

This update is performed on a logarithmic domain to ensure that the updated covariance matrices are positive definite. Although only the case of using a single observation sequence for training data is described here, it is straightforward to extend this training algorithm to multiple observation sequences. This training algorithm enables the HMM parameters to be optimized so that both the generated parameter trajectory and its GV are close to the natural ones.

The proposed objective function $\mathcal{L}'_{\boldsymbol{q}}$ is also used in parameter generation. The static feature vector sequence is determined by

$$\begin{aligned}\hat{\boldsymbol{c}} &= \arg\max_{\boldsymbol{c}} \mathcal{L}_{\boldsymbol{q}}^{(GV)'} \\ &= \arg\max_{\boldsymbol{c}} \mathcal{N}(\boldsymbol{c}; \overline{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}) \mathcal{N}(\boldsymbol{v}(\boldsymbol{c}); \boldsymbol{v}(\overline{\boldsymbol{c}}_{\boldsymbol{q}}), \boldsymbol{U}_v) \\ &= \overline{\boldsymbol{c}}_{\boldsymbol{q}}.\end{aligned} \tag{53}$$

[2] A suboptimum HMM state sequence may be determined by the Viterbi algorithm with the traditional likelihood, *i.e.*, $P(\boldsymbol{q}|\boldsymbol{\lambda})P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda})$.

Note that this estimate is equivalent to the ML estimate (given by Eq. (20)) in the traditional parameter generation algorithm because the GV likelihood is always maximized by setting the generated parameters to the mean vector of the trajectory HMM. Therefore, it is no longer necessary to practically consider the GV term in parameter generation, and the traditional parameter generation algorithm can be directly employed. Note that the traditional algorithm is computationally much more efficient than the parameter generation algorithm considering the GV, which requires an iterative process as mentioned in **Section 3.2**.

4.3 Discussion

GV-constrained trajectory training addresses some of the issues of parameter generation considering the GV described in **Section 3.4**. It provides a unified framework using the same objective function in both training and synthesis processes. It also allows the closed-form solution to be used in parameter generation. This makes it possible to implement the recursive estimation process, which generates the static feature vectors frame by frame (Tokuda et al. (1995)), which is very effective for achieving a low-delay synthesis process. Moreover, context-dependent GV modeling can be easily implemented without increasing the number of model parameters. It has been found that the variations of the GV tend to be larger with decreasing number of frames (*i.e.*, the total duration of an utterance). Therefore, the parameter generation algorithm considering the GV sometimes causes highly artificial sounds in synthetic speech when synthesizing very short utterances such as one word. This problem is effectively addressed by GV-constrained trajectory training.

In an attempt to apply the idea of considering the GV in the HMM training process, Wu *et al*. (Wu et al. (2008)) proposed MGE training that considers the error in the GV between natural and generated parameters as well as the generation error mentioned above. The HMM parameters are optimized so that both the generated parameters and their GV are similar to the natural ones. This method is similar to GV-constrained trajectory training. One of the differences between these two methods is that not only frame-by-frame (weighted) generation errors but also the correlation between the errors over a time sequence is considered in GV-constrained trajectory training because of the use of the temporal covariance matrix of the trajectory HMM, $\boldsymbol{P}_{\boldsymbol{q}}$.

5. Experimental evaluation

The effectiveness of parameter generation considering the GV and that of GV-constrained trajectory training were evaluated separately.

5.1 Experimental conditions

The 0^{th} through 24^{th} mel-cepstral coefficients were used as spectral parameters and log-scaled F_0 was used as the excitation parameter. A high-quality speech analysis-synthesis method called Speech Transformation and Representation using Adaptive Interpolation of weiGHTed spectrum (STRAIGHT) (Kawahara et al. (1999)) was employed for the analysis-synthesis method. Each speech parameter vector included the static features and their delta and delta-deltas. The frame shift was set to 5 ms.

Context-dependent phoneme HMMs were trained for each of the spectral and F_0 components using a decision-tree-based context-clustering technique based on the minimum description length (MDL) criterion (Shinoda & Watanabe (2000)). The spectral component was modeled by the continuous density HMM, of which each state output p.d.f. was modeled by a single Gaussian with a diagonal covariance matrix. The F_0 component was modeled by the multispace probability distribution HMM (MSD-HMM) (Tokuda et al. (2002)) to model a time

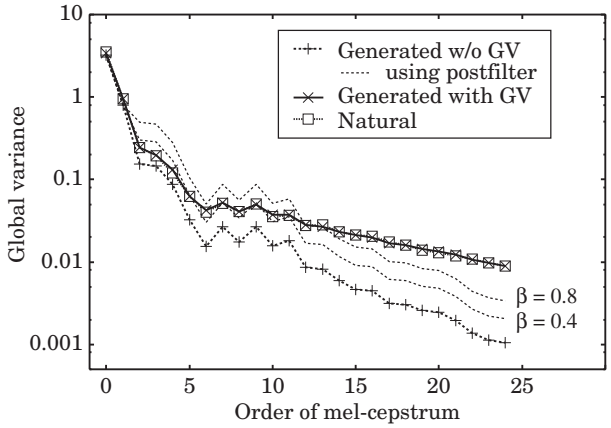


Fig. 6. GVs of several mel-cepstrum sequences. The values shown are GV mean values over all test sentences and speakers.

sequence consisting of continuous values, *i.e.*, log-scaled values of F_0, and discrete symbols that represent unvoiced frames. Static, delta, and delta-delta values of F_0 were modeled in different streams (Yoshimura et al. (1999)). Context-dependent duration models for modeling the state duration probabilities were also trained.
In the synthesis, a sentence HMM for given input contexts was constructed by concatenating the context-dependent phoneme HMMs, and then a suboptimum state sequence was determined from the state duration model. Mel-cepstrum and F_0 sequences were directly generated from p.d.f. sequences corresponding to the determined suboptimum state sequence. A speech waveform was synthesized by filtering the excitation signal, which was designed using the generated excitation parameters, based on the generated mel-cepstra with the Mel Log Spectrum Approximation (MLSA) filter (Imai (1983)).

5.2 Evaluations of parameter generation considering GV

To evaluate the parameter generation algorithm considering the GV, voices were built for four Japanese speakers (two males, MHT and MYI, and two females, FTK and FYM) in the ATR Japanese speech database B-set (Sagisaka et al. (1990)), which consists of 503 phonetically balanced sentences. For each speaker, 450 sentences were used as training data and the other 53 sentences were used for evaluation. Context-dependent labels were prepared from phoneme and linguistic labels included in the ATR database. A Gaussian distribution of the GV p.d.f. for each of the spectral and F_0 components was trained using the GVs calculated from individual utterances in the training data. The GV weight ω was set to 1.0.
Figure 6 shows the GVs of mel-cepstra generated with the traditional generation algorithm and those with the generation algorithm considering the GV. For the traditional algorithm, the GVs of the generated mel-cepstra when using the postfilter to emphasize the mel-cepstra (Koishida et al. (1995)) are also shown. The filtering coefficient β was set to 0.4 or 0.8. The GV of the natural mel-cepstra is also shown in the figure as a reference. It can be seen that the GV of the mel-cepstra generated with the traditional algorithm is small. Although postfiltering increases the GV, the GV characteristics of the emphasized mel-cepstra are obviously different from those of the natural ones. On the other hand, mel-cepstra whose GV is almost equal to that of the natural ones are generated when considering the GV.

Generation method	MOS $\pm$ 95% confidence interval
Generated w/o GV	2.53 ± 0.12
Generated with GV	3.46 ± 0.15
Natural (analysis-synthesized)	4.35 ± 0.12

Table 1. Mean opinion score (MOS) on naturalness given in opinion test to evaluate parameter generation with GV.

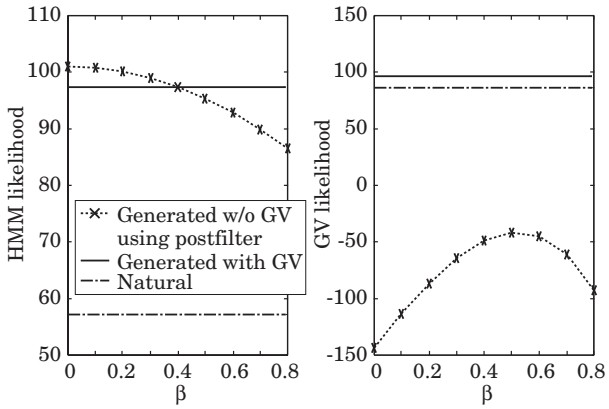


Fig. 7. Log-scaled HMM and GV likelihoods on mel-cepstrum sequences as a function of postfilter coefficient β. The HMM likelihoods are normalized by the number of frames.

Figure 7 shows the logarithmic HMM and GV likelihoods on mel-cepstrum sequences, which are normalized by the number of frames. It is reasonable that the largest HMM likelihood is yielded by the traditional algorithm ($\beta = 0.0$) and that it decreases when the postfilter is applied or the GV is considered. An interesting point is that the HMM likelihood for the natural sequence is smaller than those for the generated sequences. This implies that we do not necessarily generate the speech parameter sequence that maximizes only the HMM likelihood, although it seems reasonable to keep the likelihood larger than that for the natural sequence. The GV likelihoods are very small when using the traditional algorithm because of the GV reduction shown in **Figure 6**. Although it is observed that they are recovered by postfiltering, the resulting likelihoods are still much smaller than that for the natural sequence. On the other hand, the algorithm considering the GV generates a sequence for which the GV likelihood is sufficiently large. Consequently, it makes both HMM and GV likelihoods exceed those for the natural sequence. These results demonstrate that the algorithm considering the GV is capable of generating more similar speech parameter sequences to those of natural speech from the viewpoint of satisfying a greater variety of characteristics than the traditional algorithm.

Table 1 shows the result of a subjective evaluation based on an opinion test on the naturalness of the synthetic speech. The opinion score was set to a 5-point scale (5: excellent, 4: good, 3: fair, 2: poor, 1: bad) and ten Japanese listeners participated in the test. It is observed that the generation algorithm considering the GV yields significant quality improvements compared with the traditional generation algorithm. It effectively reduces muffled sounds of synthetic

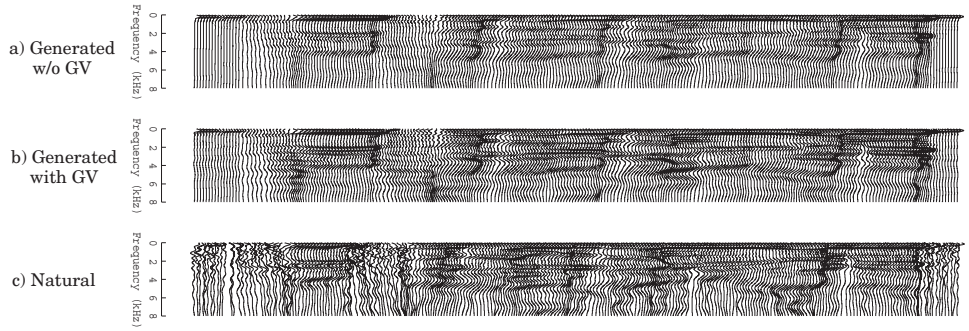


Fig. 8. Examples of spectrum sequences of generated speech using traditional algorithm, generated speech using generation algorithm considering GV, and natural speech. Note that the phoneme duration of the natural sequence is different from those of the generated sequences.

voices caused by the oversmoothing effect.[3] An example of spectrum sequences is shown in **Figure 8**. The algorithm considering the GV generates much sharper spectral peaks than those generated by the conventional algorithm. Note that increasing the GV usually causes an increase in mel-cepstral distortion between the generated sequence and the natural sequence, which is strongly correlated with the decrease in the HMM likelihood shown in **Figure 7**. It is possible that simply increasing the GV will cause the quality degradation of synthetic speech because it does not always make the generated sequence close to the natural one. The GV-based generation algorithm increases the GV by considering the GV likelihood while also considering the HMM likelihood to reduce the quality degradation due to the excessive increase in the GV.

5.3 Evaluation of GV-constrained trajectory training

To evaluate the GV-constrained trajectory training method, voices were built for four English speakers (2 males: bdl and rms, and 2 females: clb and slt) in the CMU ARCTIC database (Kominek & Black (2003)). For each speaker, we used subset A, which consists of about 600 sentences as training data and the remaining subset B, which consists of about 500 sentences for evaluation. Context-dependent labels were automatically generated from texts using a text analyzer. After initializing the HMM parameters in the traditional training process, trajectory training was performed for the spectral and F_0 components. Finally, GV-constrained trajectory training was performed for both components. The covariance matrix of the GV p.d.f. for each component was previously trained using the GVs calculated from individual utterances in the training data. The GV weight ω was empirically set to 0.125.

Table 2 shows the log-scaled trajectory likelihood, GV likelihood, and total likelihood for the training data and evaluation data, where the total likelihood is calculated as the product of the trajectory and GV likelihoods. All values are normalized by the number of frames. The trajectory training causes significant improvements in the trajectory likelihoods because the HMM parameters are optimized so as to directly maximize the trajectory likelihoods. It is interesting to note that the trajectory training also causes improvements in the GV likelihood, although the improvements are not large. These results suggest that the trajectory training

[3] Several samples are available from http://isw3.naist.jp/~tomoki/INTECH/ModelGV/index.html

Training method	Training data			Evaluation data		
	Trajectory	GV	Total	Trajectory	GV	Total
Traditional	19.06	-67.63	-48.57	16.26	-67.86	-51.60
Trajectory	30.78	-32.35	-1.57	29.30	-33.33	-4.03
GV-constrained	30.36	94.98	125.34	28.89	82.51	111.40

Table 2. Log-scaled trajectory likelihood given by Eq. (41), GV likelihood given by Eq. (44), and total likelihood given by Eq. (43) ($\omega = 1.0$) for each training method.

Training method	MOS $\pm$ 95% confidence interval
Traditional	2.35 ± 0.14
Trajectory	2.79 ± 0.12
GV-constrained	3.46 ± 0.15
Natural (analysis-synthesized)	4.38 ± 0.13

Table 3. Mean opinion score (MOS) on naturalness given in opinion test to evaluate GV-constrained trajectory training.

generates better parameter trajectories than the traditional training. The GV likelihoods are dramatically improved by GV-constrained trajectory training. Note that this training method does not cause significant reductions to the trajectory likelihoods. This can be observed in both the training and evaluation data. Overall, these results suggest that the GV-constrained trajectory training method leads to parameter trajectories that more closely resemble the various characteristic features of real speech.

Table 3 shows the result of a subjective evaluation based on an opinion test on the naturalness of the synthetic speech. The opinion score was set to the same 5-point scale as before and ten listeners participated in the test. GV-constrained trajectory training yields significant quality improvements compared with the traditional training. This tendency is similar to that observed when considering the GV in the parameter generation process, as shown in **Table 1**. The trajectory training also yields significant quality improvements, but these improvements are much smaller than those yielded by GV-constrained trajectory training. [4]

6. Summary

This chapter has described the two main techniques for modeling a speech parameter sequence considering the global variance (GV) for HMM-based speech synthesis: the parameter generation algorithm considering the GV and GV-constrained trajectory training. The traditional framework of HMM-based speech synthesis suffers from the oversmoothing effect in speech parameters generated from the HMM, which makes the synthetic speech sound muffled. Since the GV is inversely correlated with the oversmoothing effect, a metric on the GV of the generated parameters is effectively used to reduce muffled sounds. The parameter generation algorithm considering the GV uses not only an HMM likelihood but also a GV likelihood to determine the generated parameters. GV-constrained trajectory training has been proposed by integrating this idea into a training framework. Consequently, it provides a unified framework for training and synthesizing speech using a common criterion, context-dependent GV modeling, and a more efficient parameter generation process with the GV based on a closed-form solution. Experimental results have demonstrated that both

[4] Several samples are available from http://isw3.naist.jp/~tomoki/INTECH/ModelGV/index.html

methods yield very significant improvements in the naturalness of synthetic speech and that GV modeling is very effective in HMM-based speech synthesis.

Acknowledgements: This research was supported in part by MEXT Grant-in-Aid for Young Scientists (A).

7. References

Donovan, R. & Woodland, P. Improvements in an HMM-based speech synthesiser. *Proceedings of Euro. Conf. on Speech Commun. & Tech. (EUROSPEECH)* pp. 573–576, Madrid, Spain, Sept. 1995.

Gales, M. & Young, S. The application of hidden Markov models in speech recognition. *Foundations & Trends in Signal Processing* Vol. 1, No. 3, pp. 195–304, 2008.

Huang, X.; Acero, A.; Adcock, J.; Hon, H.-W.; Goldsmith, J.; Liu, J. & Plumpe, M. Whistler: a trainable text-to-speech system. *Proceedings of Int. Conf. on Spoken Lang. Process. (ICSLP)*, pp. 2387–2390, Philadelphia, USA, Oct. 1996.

Imai, S. Cepstral analysis synthesis on the mel frequency scale. *Proceedings of IEEE Int. Conf. on Acoust., Speech, & Signal Process. (ICASSP)*, pp. 93–96, Boston, USA, Apr. 1983.

Kawahara, H.; Masuda-Katsuse, I. & de Cheveigné, A. Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F_0 extraction: possible role of a repetitive structure in sounds. *Speech Commun.*, Vol. 27, No. 3–4, pp. 187–207, 1999.

Koishida, K.; Tokuda, K.; Kobayashi, T. & Imai, S. CELP coding based on mel-cepstral analysis. *Proceedings of IEEE Int. Conf. on Acoust., Speech, & Signal Process. (ICASSP)*, pp. 33–36, Detroit, USA, May 1995.

Kominek, J. & Black, A.W. CMU ARCTIC databases for speech synthesis. *Technical Report*, CMU-LTI-03-177, Language Technologies Institute, Carnegie Mellon University, 2003.

Sagisaka, Y.; Takeda, K.; Abe, M.; Katagiri, S.; Umeda, T. & Kuwabara, H. A large-scale Japanese speech database. *Proceedings of Int. Conf. on Spoken Lang. Process. (ICSLP)*, pp. 1089–1092, Kobe, Japan, Nov. 1990.

Shinoda, K. & Watanabe, T. MDL-based context-dependent subword modeling for speech recognition. *J. Acoust. Soc. Jpn. (E)*, Vol. 21, No. 2, pp. 79–86, 2000.

Toda, T. & Tokuda, K. A speech parameter generation algorithm considering global variance for HMM-based speech synthesis. *IEICE Trans. Inf. & Syst.*, Vol. E90-D, No. 5, pp. 816–824, 2007.

Toda, T.; Black, A.W. & Tokuda, K. Voice conversion based on maximum likelihood estimation of spectral parameter trajectory. *IEEE Trans. Audio, Speech & Lang. Process.*, Vol. 15, No. 8, pp. 2222–2235, 2007.

Toda, T. & Young, S. Trajectory training considering global variance for HMM-based speech synthesis. *Proceedings of IEEE Int. Conf. on Acoust., Speech, & Signal Process. (ICASSP)*, pp. 4025–4028, Taipei, Taiwan, Apr. 2009.

Tokuda, K.; Kobayashi, T.; Masuko, T. & Imai, S. Mel-generalized cepstral analysis – a unified approach to speech spectral estimation. *Proceedings of Int. Conf. on Spoken Lang. Process. (ICSLP)*, pp. 1043–1045, Yokohama, Japan, Sept. 1994.

Tokuda, K.; Kobayashi, T. & Imai, S. Speech parameter generation from HMM using dynamic features. *Proceedings of IEEE Int. Conf. on Acoust., Speech, & Signal Process. (ICASSP)*, pp. 660–663, Detroit, USA, May 1995.

Tokuda, K.; Yoshimura, T.; Masuko, T.; Kobayashi, T. & Kitamura, T. Speech parameter generation algorithms for HMM-based speech synthesis. *Proceedings of IEEE Int. Conf.*

on Acoust., Speech, & Signal Process. (ICASSP), pp. 1315–1318, Istanbul, Turkey, June 2000.

Tokuda, K.; Masuko, T.; Miyazaki, N. & Kobayashi, T. Multi-space probability distribution HMM. *IEICE Trans. Inf. & Syst.*, Vol. E85-D, No. 3, pp. 455–464, 2002.

Wu, Y.-J. & Wang, R.H. Minimum generation error training for HMM-based speech synthesis. *Proceedings of IEEE Int. Conf. on Acoust., Speech, & Signal Process. (ICASSP)*, pp. 89–92, Toulouse, France, May 2006.

Wu, Y.-J.; Zen, H.; Nankaku, Y. & Tokuda, K. Minimum generation error criterion considering global/local variance for HMM-based speech synthesis. *Proceedings of IEEE Int. Conf. on Acoust., Speech, & Signal Process. (ICASSP)*, pp. 4621–4624, Las Vegas, USA, Mar. 2008.

Yamagishi, J.; Nose, T.; Zen, H.; Ling, Z.-H.; Toda, T.; Tokuda, K.; King, S. & Renals, S. Robust speaker-adaptive HMM-based text-to-speech synthesis. *IEEE Trans. Audio, Speech & Lang. Process.*, Vol. 17, No. 6, pp. 1208–1230, 2009.

Yoshimura, T.; Tokuda, K.; Masuko, T.; Kobayashi, T. & Kitamura, T. Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. *Proceedings of Euro. Conf. on Speech Commun. & Tech. (EUROSPEECH)* pp. 2347–2350, Budapest, Hungary, Sept. 1999.

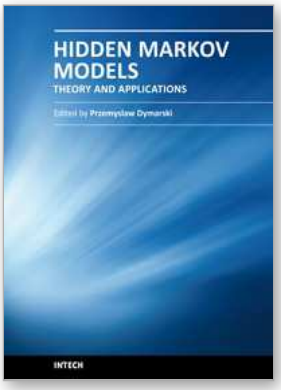
Zen, H.; Toda, T.; Nakamura, M. & Tokuda, K. Details of the Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005. *IEICE Trans. Inf. & Syst.*, Vol. E90-D, No. 1, pp. 325–333, 2007a.

Zen, H.; Tokuda, K.; Masuko, T.; Kobayashi, T. & Kitamura, T. A hidden semi-Markov model-based speech synthesis system. *IEICE Trans. Inf. & Syst.*, Vol. E90-D, No. 5, pp. 825–834, 2007b.

Zen, H.; Tokuda, K. & Kitamura, T. Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences. *Computer Speech & Lang.*, Vol. 21, No. 1, pp. 153–173, 2007c.

Zen, H.; Toda, T. & Tokuda, K. The Nitech-NAIST HMM-based speech synthesis system for the Blizzard Challenge 2006. *IEICE Trans. Inf. & Syst.*, Vol. E91-D, No. 6, pp. 1764–1773, 2008.

Zen, H.; Tokuda, K. & Black, A.W. Statistical parametric speech synthesis. *Speech Commun.*, Vol. 51, No. 11, pp. 1039–1064, 2009.



Hidden Markov Models, Theory and Applications

Edited by Dr. Przemyslaw Dymarski

ISBN 978-953-307-208-1
Hard cover, 314 pages
Publisher InTech
Published online 19, April, 2011
Published in print edition April, 2011

Hidden Markov Models (HMMs), although known for decades, have made a big career nowadays and are still in state of development. This book presents theoretical issues and a variety of HMMs applications in speech recognition and synthesis, medicine, neurosciences, computational biology, bioinformatics, seismology, environment protection and engineering. I hope that the reader will find this book useful and helpful for their own research.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Tomoki Toda (2011). Modeling of Speech Parameter Sequence Considering Global Variance for HMM-Based Speech Synthesis, Hidden Markov Models, Theory and Applications, Dr. Przemyslaw Dymarski (Ed.), ISBN: 978-953-307-208-1, InTech, Available from: http://www.intechopen.com/books/hidden-markov-models-theory-and-applications/modeling-of-speech-parameter-sequence-considering-global-variance-for-hmm-based-speech-synthesis

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821