社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

Reformulating the HMM as a Trajectory Model

Keiichi TOKUDA†, Heiga ZEN†, and Tadashi KITAMURA†

† Graduate School of Engineering, Nagoya Institute of Technology
Gokiso-cho, Showa-ku, Nagoya, 466–8555 Japan

Abstract We have shown that the HMM whose state output vector includes static and dynamic feature parameters can be reformulated as a trajectory model by imposing the explicit relationship between the static and dynamic features. The derived model, referred to as "trajectory HMM," can alleviate the limitations of HMMs: i) constant statistics within an HMM state and ii) independence assumption of state output probabilities. In this paper, we first summarize the definition and the training algorithm. Then, to show that the trajectory HMM is a proper generative model, we derive a new algorithm for sampling from the trajectory model, and show the result of an illustrative experiment. A speech recognition experiment demonstrates the consistency between training and decoding criteria is essential: the model should not only be traind as a trajectory model but also be used as a trajectory model in decoding, even though the trajectory model has the same parameterization as the standard HMM.

Key words HMM, speech recognition, speech synthesis, trajectory model, dynamic feature

1. Introduction

Tractable and efficient implementations of the HMM framework are based on assumptions: i) piece-wise constant statistics within an HMM state and ii) independence assumption of state output probabilities. To overcome these limitations, alternative models have been proposed, e.g., [1]– [10]. Most of them have attempted to capture the explicit dynamics of speech parameter trajecotories. The use of the dynamic features (delta and delta-delta features) [11] can also enhance the performance of HMM-based speech recognizers. However, it has been thought of as an ad hoc rather than an essential solution: the standard HMM allows inconsistent static and dynamic features when it is used as a generative model. On the other hand, we have shown that by imposing the explicit relationship between static features and dynamic features on the standard HMM, it is naturally translated into a trajectory model, referred to as "trajectory HMM" [12]. We have also shown that the effectiveness of the trajectory HMM in speech recognition [13] and speech synthesis [14].

In this paper, we first summarize the definition and algorithms of the trajectory HMM. Then, we demonstrate that the trajectory HMM is a proper generative model, by showing an illustrative experiment with a new algorithm for sampling from the trajectory model. A speech recognition experiment is also conducted to demonstrates the importance of the consistency between training and decoding criteria: not only the model should be traind as a trajectory model but also it should be used as a trajectory model in decoding, even though the trajectory model has the same parameterization as the standard HMM.

The formulation of the trajectory HMM is closely related to a technique for parameter generation from HMM [15]– [17], in which the speech parameter sequence is determined so as to maximize its output probability for the HMM under the constraints between static and dynamic features. While we derived the speech parameter generation algorithm in order to construct HMM-based speech synthesizers [18] which can synthesize speech with various voice characteristics, the generation algorithm was also applied to speech recognition in [19]. Interesting discussions related to the use of dynamic features in HMMs can be found in [20], [21]. This paper also discusses relations between the trajectory HMM and other techniques.

The rest of the paper organized as follows. Section 2 describes the definition of the trajectory model. Section 3 shows the relation to the HMM-based speech synthesis approach. Section 4 summarizes the training algorithm. A sampling algorithm is derived in Section 5. In Section 6, the result of a speech recogniton experiment shows the importance of consistency between training and decoding. Concluding remarks and future plans are given in the final section.

2. Refomulating HMM

The output probability of a speech parameter vector sequence $\boldsymbol{o} = \left[\boldsymbol{o}_1^\top, \boldsymbol{o}_2^\top, \ldots, \boldsymbol{o}_T^\top\right]^\top$ for the standard HMM is given by

$$P(\boldsymbol{o} \mid \lambda) = \sum_{\text{all } \boldsymbol{q}} P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda) P(\boldsymbol{q} \mid \lambda), \tag{1}$$

where $\boldsymbol{q} = \{q_1, q_2, \ldots, q_T\}$ is a state sequence. In most of speech recognition systems, the speech parameter vector $\boldsymbol{o}_t$ is assumed to consist of the static feature vector $\boldsymbol{c}_t = [c_t(1), c_t(2), \ldots, c_t(M)]^\top$ (e.g., cepstral coefficients),

and dynamic feature vectors $\Delta \boldsymbol{c}_t$, $\Delta^2 \boldsymbol{c}_t$ (e.g., delta and delta-delta cepstral coefficients), that is $\boldsymbol{o}_t = [\boldsymbol{c}_t^\top, \Delta \boldsymbol{c}_t^\top, \Delta^2 \boldsymbol{c}_t^\top]^\top$. The dynamic features calculated by

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

$$\Delta^2 \boldsymbol{c}_t = \sum_{\tau=-L}^{L} w^{(2)}(\tau) \boldsymbol{c}_{t+\tau}. \tag{3}$$

correspond to the first and second time-derivative of the static feature $\boldsymbol{c}_t$, respectively, where $\left\{w^{(n)}(\tau)\right\}_{\tau=-L}^{L}$ are the coefficients for calculating the n-th dynamic feature and usually L is around from 1 to 3. Conditions (2) and (3) can be arranged in a matrix form:

$$\boldsymbol{o} = \boldsymbol{W} \boldsymbol{c}, \tag{4}$$

where $\boldsymbol{c} = \left[\boldsymbol{c}_1^\top, \boldsymbol{c}_2^\top, \ldots, \boldsymbol{c}_T^\top\right]^\top$, $\boldsymbol{W}$ is a sparse matrix given by

$$\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_T]^\top \otimes \boldsymbol{I}_{M \times M} \tag{5}$$

$$\boldsymbol{w}_t = \left[\boldsymbol{w}_t^{(0)}, \boldsymbol{w}_t^{(1)}, \boldsymbol{w}_t^{(2)}\right] \tag{6}$$

$$\boldsymbol{w}_t^{(n)} = \big[\underbrace{0, \ldots, 0}_{t-L-1}, \underbrace{w^{(n)}(-L), \ldots, w^{(n)}(0), \ldots, w^{(n)}(L)}_{2L+1}, \underbrace{0, \ldots, 0}_{T-(t+L)}\big]^\top, \quad n = 0, 1, 2, \tag{7}$$

and

$$w^{(0)}(\tau) = \begin{cases} 1, & \tau = 0 \\ 0, & \text{otherwise} \end{cases}. \tag{8}$$

When each state output probability distribution is assumed to be single Gaussian, $P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda)$ is given by

$$P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{o}_t \mid \boldsymbol{\mu}_{q_t}, \boldsymbol{\Sigma}_{q_t}) = \mathcal{N}(\boldsymbol{o} \mid \boldsymbol{\mu}_{\boldsymbol{q}}, \boldsymbol{\Sigma}_{\boldsymbol{q}}), \tag{9}$$

where $\boldsymbol{\mu}_{q_t}$ and $\boldsymbol{\Sigma}_{q_t}$ are the $3M \times 1$ mean vector and the $3M \times 3M$ covariance matrix, respectively, associated with q_t-th state, and

$$\boldsymbol{\mu}_{\boldsymbol{q}} = \left[\boldsymbol{\mu}_{q_1}^\top, \boldsymbol{\mu}_{q_2}^\top, ..., \boldsymbol{\mu}_{q_T}^\top\right]^\top \tag{10}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{q}} = \mathrm{diag}\left[\boldsymbol{\Sigma}_{q_1}, \boldsymbol{\Sigma}_{q_2}, ..., \boldsymbol{\Sigma}_{q_T}\right]. \tag{11}$$

The above model is improper in the sense of statistical modeling: it allows inconsistent static and dynamic feature vector sequences even though they are constrained by (4). To avoid the problem, the statistical model should be defined as a function of $\boldsymbol{c}$ because the original observation is $\boldsymbol{c}$ rather than the augmented variable $\boldsymbol{o}$. Accordingly, $P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda)$ should be normalized by the normalization term $K_{\boldsymbol{q}}$:

$$P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda) = \frac{1}{K_{\boldsymbol{q}}} P(\boldsymbol{K}\boldsymbol{c} \mid \boldsymbol{q}, \lambda), \tag{12}$$

where

$$K_{\boldsymbol{q}} = \int P(\boldsymbol{K}\boldsymbol{c} \mid \boldsymbol{q}, \lambda)\, d\boldsymbol{c} \tag{13}$$

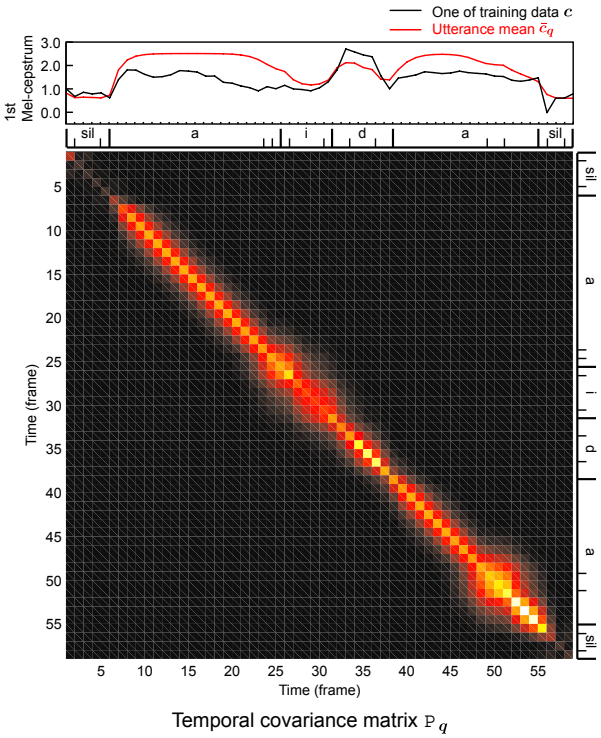

Figure 1 Mean trajectory $\boldsymbol{c}_{\boldsymbol{q}}$ and covariance matrix $\boldsymbol{P}_{\boldsymbol{q}}$.

By substituting (4) for (9), we can rewrite (9) as follows:

$$\begin{aligned} P(\boldsymbol{W}\boldsymbol{c} \mid \boldsymbol{q}, \lambda) &= \mathcal{N}(\boldsymbol{W}\boldsymbol{c} \mid \boldsymbol{\mu}_{\boldsymbol{q}}, \boldsymbol{\Sigma}_{\boldsymbol{q}}) \\ &= K_{\boldsymbol{q}} \cdot \mathcal{N}(\boldsymbol{c} \mid \overline{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}), \end{aligned} \tag{14}$$

where $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ is given by

$$\boldsymbol{R}_{\boldsymbol{q}} \overline{\boldsymbol{c}}_{\boldsymbol{q}} = \boldsymbol{r}_{\boldsymbol{q}} \tag{15}$$

and

$$\boldsymbol{R}_{\boldsymbol{q}} = \boldsymbol{W}^\top \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \boldsymbol{W} \tag{16}$$

$$\boldsymbol{r}_{\boldsymbol{q}} = \boldsymbol{W}^\top \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \boldsymbol{\mu}_{\boldsymbol{q}} \tag{17}$$

$$\boldsymbol{P}_{\boldsymbol{q}} = \boldsymbol{R}_{\boldsymbol{q}}^{-1} \tag{18}$$

$$K_{\boldsymbol{q}} = \frac{\sqrt{(2\pi)^{MT} |\boldsymbol{P}_{\boldsymbol{q}}|}}{\sqrt{(2\pi)^{3MT} |\boldsymbol{\Sigma}_{\boldsymbol{q}}|}} \cdot \exp\left\{-\frac{1}{2}\left(\boldsymbol{\mu}_{\boldsymbol{q}}^\top \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \boldsymbol{\mu}_{\boldsymbol{q}} - \boldsymbol{r}_{\boldsymbol{q}}^\top \boldsymbol{P}_{\boldsymbol{q}} \boldsymbol{r}_{\boldsymbol{q}}\right)\right\}. \tag{19}$$

Thus, we may redefine the standard HMM (1) as follows:

$$\begin{aligned} P(\boldsymbol{c} \mid \lambda) &= \sum_{\text{all } \boldsymbol{q}} \frac{1}{K_{\boldsymbol{q}}} P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda) P(\boldsymbol{q} \mid \lambda) \\ &= \sum_{\text{all } \boldsymbol{q}} P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda) P(\boldsymbol{q} \mid \lambda), \end{aligned} \tag{20}$$

where

$$P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda) = \mathcal{N}(\boldsymbol{c} \mid \overline{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}). \tag{21}$$

It is interesting to note that the mean $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ is exactly the same as the speech parameter trajectory obtained by a speech parameter generation technique for HMM-based speech synthesis which will be summarized in the next section. By assuming the parameter trajectory $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ as the mean

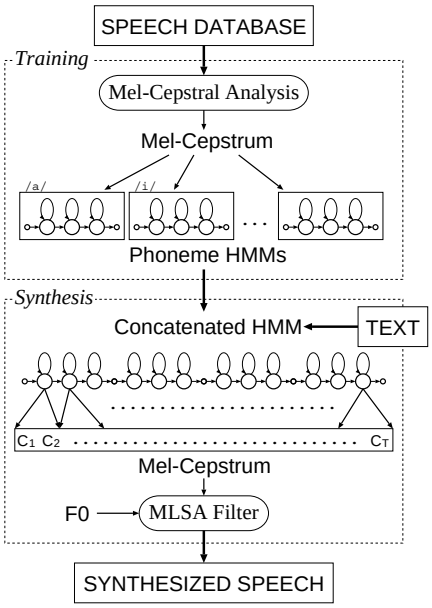

Figure 2 HMM-based speech synthesis system

for the spectral parameter vector sequence $\boldsymbol{c}$ corresponding to an utterance, the standard HMM can naturally be translated into a trajectory model: the state output probability of observing the static part of the output vector changes during a state, and is affected by statistics of neighboring states. Note that the spectral parameter vector sequence $\boldsymbol{c}$ is modeled by a mixture of Gaussians whose dimensionality is TM, and their covariances $\boldsymbol{P}_{\boldsymbol{q}}$ are generally full. As a result, the trajectory HMM can alleviate the dificiency of the standard HMM. It is also noted that the prametrization of the trajectory HMM is exactly the same as the standard HMM: the number of parameters of the trajectory HMM is the same as the standard HMM with the same topology.

Figure 1 shows an example of the mean trajectory $\boldsymbol{c}_{\boldsymbol{q}}$ and the covariance matrix $\boldsymbol{P}_{\boldsymbol{q}}$. Model training conditions are the same as those in Section 6. To obtain a state sequence $\boldsymbol{q}$, a concatenated model composed of phoneme models /sil/, /a/, /i/, /d/, /a/, /sil/ was aligned to a test utterance using the modified Viterbi algorithm derived in [13]. Note that only elements corresponding to the first coefficient of melcepstrum are shown in the figure. It can be seen that not only the mean trajectory $\boldsymbol{c}_{\boldsymbol{q}}$ varies in each state but also the temporal correlation can be modeled by the covariance $\boldsymbol{P}_{\boldsymbol{q}}$. It is also interesting to note that the mean trajectory and the temporal covariance corresponding each monophone model vary according to its state durations and neighoring models (see phoneme /a/). This shows that the trajectory model has the capability to capture the coarticulation effects naturally.

3. Relation to HMM-based speech synthesis

Figure 2 shows the block diagram of an HMM-based speech synthesis sytem. In the training part, spectrum parameters are extracted from a speech database and modeled by context-dependent phoneme HMMs. In the synthesis part, context-dependent HMMs are concatenated according to the text to be synthesized. Then, spectrum parameters are generated from the HMM by using a speech-parameter-generation algorithm [15]– [17]. Finally, the synthesis-filter module synthesize the speech waveform using the generated spectrum parameters(注1). The attraction of this approach is in that voice qualities of synthesized speech can easily be changed by transforming HMM parameters. In fact, it has been shown that we can change voice qualities of synthesized speech by applying a speaker adaptation technique [22], a speaker interpolation technique [23], or an eigenvoice technique [24].

In this framework, the problem of speech synthesis can be represented by

$$\boldsymbol{o}_{\max} = \arg\max_{\boldsymbol{o}} P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda) \tag{22}$$

for a state sequence $\boldsymbol{q}$ which can be determined by state duration models. Unfortunately, without any constraints, $P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda)$ is maximized when $\boldsymbol{o} = \boldsymbol{\mu}_{\boldsymbol{q}}$, that is, the speech parameter vector sequence becomes a sequence of the mean vectors. To avoid this, we apply the relationship between static and dynamic features (4) as constraints of the maximization problem. With these constraints, maximizing $P(\boldsymbol{o} \mid \boldsymbol{q}, \lambda)$ with respect to $\boldsymbol{o}$ is equivalent to that with respect to $\boldsymbol{c}$:

$$\boldsymbol{c}_{\max} = \arg\max_{\boldsymbol{c}} P(\boldsymbol{W}\boldsymbol{c} \mid \boldsymbol{q}, \lambda). \tag{23}$$

By setting $\partial \log P(\boldsymbol{W}\boldsymbol{c}_{\max} \mid \boldsymbol{q}, \lambda)/\partial \boldsymbol{c}_{\max} = \boldsymbol{0}$, we obtain a set of equations

$$\boldsymbol{R}_{\boldsymbol{q}}\boldsymbol{c}_{\max} = \boldsymbol{r}_{\boldsymbol{q}}, \tag{24}$$

where $\boldsymbol{R}_{\boldsymbol{q}}$ and $\boldsymbol{r}_{\boldsymbol{q}}$ are given by (16) and (17), respectively. Since (24) coincides with (15), the generated speech parameter sequence $\boldsymbol{c}$ is identical with the mean trajectory $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ of the trajectory model.

For direct solution of (24), we need $O(T^3M^3)$ operations because $\boldsymbol{R}_{\boldsymbol{q}}$ is a $TM \times TM$ matrix. By utilizing the special structure of band symmetric matrix $\boldsymbol{R}_{\boldsymbol{q}}$, (24) can be solved by the Cholesky decomposition with $O(TM^3L^2)$ operations.

The matrix $\boldsymbol{R}_{\boldsymbol{q}}$ becomes a $(4ML+1)$-diagonal symmetric positive definite matrix. Thus, $\boldsymbol{R}_{\boldsymbol{q}}$ can be decomposed by the Cholesky decomposition:

$$\boldsymbol{R}_{\boldsymbol{q}} = \boldsymbol{U}_{\boldsymbol{q}}^{\top}\boldsymbol{U}_{\boldsymbol{q}}, \tag{25}$$

where $\boldsymbol{U}_{\boldsymbol{q}}$ is a upper $(2ML+1)$-diagonal triangular matrix. Then, the set of equations (24) can be rewritten by the following two set of equations:

$$\boldsymbol{U}_{\boldsymbol{q}}^{\top}\boldsymbol{g}_{\boldsymbol{q}} = \boldsymbol{r}_{\boldsymbol{q}} \tag{26}$$

(注1) : We have extended the system so as to model spectrum parameters, fundamental frequency (F0) parameters, and durations simultaneously in a unified framework [18].

$$\boldsymbol{U}_{\boldsymbol{q}}\overline{\boldsymbol{c}}_{\boldsymbol{q}} = \boldsymbol{g}_{\boldsymbol{q}}, \tag{27}$$

where (26) and (27) can be solved by the forward substitution and the backward substitution, respectively. As a result, we can compute the solution $\overline{\boldsymbol{c}}_{\boldsymbol{q}}$ with $O(TM^3L^2)$, which is reduced to $O(TML^2)$ when $\boldsymbol{\Sigma}_q$ is diagonal because each of the M-dimensions can be calculated independently(注2).

Equation (24) can also be solved by an algorithm derived in [15]–[17], which can operate in a time-recursive manner [26].

4. Training Algorithm

In this section, we summarize a training algorithm for the trajectory model. It should be noted that although the model has the same parameterization as the standard HMM, the output probability is defined by (20) rather than by (1). Accordingly, the model parameters should be trained based on (20).

An auxiliary function of current parameter set λ and new parameter set λ' is defined by

$$\mathcal{Q}(\lambda, \lambda') = \sum_{\text{all } \boldsymbol{q}} P(\boldsymbol{q} \mid \boldsymbol{c}, \lambda) \log P(\boldsymbol{c}, \boldsymbol{q} \mid \lambda'). \tag{28}$$

Although it can be shown that by substituting λ' which maximizes $\mathcal{Q}(\lambda, \lambda')$ for λ, the likelihood increases unless λ is a critical point of the likelihood, we apply the single Viterbi path approximation(注3) because it is not tractable to evaluate all possible state sequences. As a result, the problem is broken down into the following maximization problems:

$$\boldsymbol{q}_{\max} = \arg\max_{\boldsymbol{q}} P(\boldsymbol{c}, \boldsymbol{q} \mid \lambda) \tag{29}$$

$$\lambda' = \arg\max_{\lambda} P(\boldsymbol{c}, \boldsymbol{q}_{\max} \mid \lambda) \tag{30}$$

It is still difficult to solve the problem of (29) by the conventional Viterbi algorithm because temporal covariance matrix $\boldsymbol{P}_{\boldsymbol{q}}$ is generally full. Thus, a modified Viterbi algorithm to obtain a "sub-optimal" state sequence is derived in [13] based on a time-recursive likelihood calculation and the delayed decision strategy. Once we obtain a sub-optimal state sequnce, we can solve the maximization problem of (30) as follows.

The problem is equivalent to maximizing

$$\log P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda) = -\frac{1}{2}\Big\{MT\log(2\pi) - \log|\boldsymbol{R}_{\boldsymbol{q}}| + \boldsymbol{c}^\top \boldsymbol{P}_{\boldsymbol{q}}^{-1}\boldsymbol{c} + \boldsymbol{r}_{\boldsymbol{q}}^\top \boldsymbol{P}_{\boldsymbol{q}}\boldsymbol{r}_{\boldsymbol{q}} - 2\,\boldsymbol{r}_{\boldsymbol{q}}^\top \boldsymbol{c}\Big\} \tag{31}$$

with respect to

$$\boldsymbol{m} = \left[\boldsymbol{\mu}_1^\top, \boldsymbol{\mu}_2^\top, \ldots, \boldsymbol{\mu}_N^\top\right]^\top \tag{32}$$

$$\boldsymbol{\phi} = \left[\boldsymbol{\Sigma}_1^{-1}, \boldsymbol{\Sigma}_2^{-1}, \ldots, \boldsymbol{\Sigma}_N^{-1}\right]^\top, \tag{33}$$

where N is the total number of Gaussians. In this paper, we refer to $P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)$ as "trajectory likelihood."

(注2) : When $L = 1$, and $w^{(2)}(i) \equiv 0$, it is reduced to $O(TM)$ as described in [25].

(注3) : We can also use the n-best approximation.

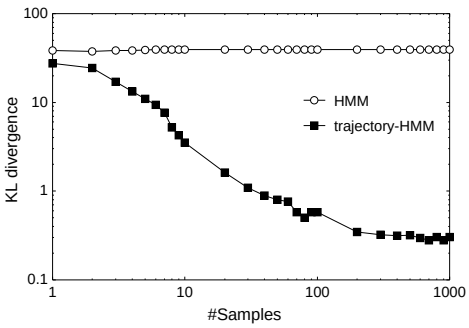

Figure 3 Convergence of the model parameters estimated from drawn samples.

By setting $\partial \log P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)/\partial \boldsymbol{m} = \boldsymbol{0}$, we obtain a set of linear equations

$$\boldsymbol{S}_{\boldsymbol{q}}^\top \boldsymbol{W}\boldsymbol{P}_{\boldsymbol{q}}\boldsymbol{W}^\top \boldsymbol{S}_{\boldsymbol{q}}\boldsymbol{\Phi}\boldsymbol{m} = \boldsymbol{S}_{\boldsymbol{q}}^\top \boldsymbol{W}\boldsymbol{c} \tag{34}$$

for determination of $\boldsymbol{m}$ which maximizes $\log P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)$, where

$$\boldsymbol{\Phi} = \operatorname{diag}(\boldsymbol{\phi}) \tag{35}$$

$$\boldsymbol{\mu}_{\boldsymbol{q}} = \boldsymbol{S}_{\boldsymbol{q}}\boldsymbol{m} \tag{36}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} = \operatorname{diag}(\boldsymbol{S}_{\boldsymbol{q}}\boldsymbol{\phi}), \tag{37}$$

and $\boldsymbol{S}_{\boldsymbol{q}}$ is a $3T \times 3MN$ matrix whose elements are 0 or 1 determined according to the state sequence $\boldsymbol{q}$. The dimensionality of (34) is $3MN$: although it could be tens of thousands, it is still small enough to solve the set of linear equations using currently available computational resources.

For maximizing $\log P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)$ with respect to $\boldsymbol{\phi}$, we apply a steepest descent algorithm using the first derivative

$$\frac{\partial \log P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)}{\partial \boldsymbol{\phi}} = \frac{1}{2}\boldsymbol{S}_{\boldsymbol{q}}^\top \operatorname{diag}^{-1}\big(\boldsymbol{W}\boldsymbol{P}_{\boldsymbol{q}}\boldsymbol{W}^\top - \boldsymbol{W}\boldsymbol{c}\boldsymbol{c}^\top\boldsymbol{W}^\top + 2\boldsymbol{\mu}_{\boldsymbol{q}}\boldsymbol{c}^\top\boldsymbol{W}^\top + \boldsymbol{W}\overline{\boldsymbol{c}}_{\boldsymbol{q}}\overline{\boldsymbol{c}}_{\boldsymbol{q}}^\top\boldsymbol{W}^\top - 2\boldsymbol{\mu}_{\boldsymbol{q}}\overline{\boldsymbol{c}}_{\boldsymbol{q}}^\top\boldsymbol{W}^\top\big) \tag{38}$$

because (31) is not a quadratic function of $\boldsymbol{\phi}$.

5. Sampling from the model

In this section, we describe a new algorithm for sampling from $P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)$, and show the result of an illustrative experiment to demonstrate that the trajectory HMM is a proper generative model.

We can draw sample from $P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)$ of (21), by sampling from a Gaussian $\mathcal{N}(\boldsymbol{c} \mid \overline{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}})$. However, we cannot use a straight forward way because the dimmensionality is high enough: TM. Thus, we derive a fast algorithm utilizing the Cholesky decomposition of (25)–(27). Assuming $\boldsymbol{w} = \left[\boldsymbol{w}_1^\top, \boldsymbol{w}_2^\top, \ldots, \boldsymbol{w}_T^\top\right]^\top$ is an M-dimmensional i.i.d. Gaussian process with mean 0 and variance 1, we may obtain a sample $\boldsymbol{x} = \left[\boldsymbol{x}_1^\top, \boldsymbol{x}_2^\top, \ldots, \boldsymbol{x}_T^\top\right]^\top$ by solving the following set of equatoins:

$$\boldsymbol{U}_{\boldsymbol{q}}\boldsymbol{x} = \boldsymbol{g}_{\boldsymbol{q}} + \boldsymbol{w}, \tag{39}$$

where $\boldsymbol{w}_t = [w_t(1), w_t(2), \ldots, w_t(M)]^\top$ and $\boldsymbol{x}_t =$

$[x_t(1), x_t(2), \ldots, x_t(M)]^\top$. We can easily confirm that

$$\overline{\boldsymbol{c}}_{\boldsymbol{q}} = E[\boldsymbol{x}] \tag{40}$$

$$\boldsymbol{P}_{\boldsymbol{q}} = E[\boldsymbol{x} - \overline{\boldsymbol{c}}_{\boldsymbol{q}}][\boldsymbol{x} - \overline{\boldsymbol{c}}_{\boldsymbol{q}}]^\top \tag{41}$$

The total computational complexity to draw a sample for given $\boldsymbol{\mu}_{\boldsymbol{q}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{q}}$ is $O(TM^3L^2)$ including that of the Cholesky decompositon. It can be reduced to $O(TML^2)$ when $\boldsymbol{\Sigma}_q$ is diagonal.

To demonstrate that the model $P(\boldsymbol{c} \mid \boldsymbol{q}, \lambda)$ of (21) is a proper generative model, we train model parameters using drawn samples from a given model, and confirm that the estimated model converges to the orignal model, by measuring the KL divergence between the original and the estimated. Figure 3 shows the result of the experiment. For the simplicity, we used a single-sate model. It is shown that as the number of drawn sample increases, the KL divergence decreases. On the other hand, the model trained by the algorithm for the standard HMM does not converges to the original.

6. consistency between training and decoding

In [13], we have shown that recognition error reduction over the standard HMM can be achieved by the trajectory HMM. In this section, we show the importance of the consistency between training and decoding, by combining additional experimental results with the results of [13].

Table 1 shows the phoneme recognition error rates for combinations of training and decoding criteria. In the table, "HMM" and "trajectory" denote training/decoding algorithms based on (9) and (21), respectively. To obtain recognition result for the trajectory HMM, we adopted a n-best rescoring strategy: n-best list generated by the standard baseline HMM was rescored by using the modified Viterbi algorithm of [13].

When 100-best list was used, (a) in the table, it is seen that (trajectory-training + trajectory-decoding) achieved 8.6% of error reduction. Discrepant training and decoding criteria: (HMM-training + trajectory-decoding) and (trajectory-training + HMM-decoding) did not work. Especially, the result for (trajectory-training + HMM-decoding) was suprisingly bad. These results indicate the importance of the consistency between training and decoding criteria.

When 100-best plus reference (correct phoneme string) list was used (b) in the table, we achieved about 40% of error reduction. This suggests that the decoder designed based on the trajectory likekihood could be further imporoved.

We also have seen the importance of the consistency between training and generation criteria in speech synthesis application in [14]. Since we have not implemented

7. Discussions

The trajectory-HMM is related to the recognition method proposed in [19], [27], which uses the trajectory synthesized by parameter generation algorithm with sliding window [26][注4], as mean vector sequence. Compared with this method, the trajectory-HMM does not require additional parameters to represent variances between training data and mean trajectories.

Table 1 Recognition error rates for combinations of training/decoding criteria. (a) 100-best rescoring, (b) (100-best + reference) rescoring

Training	HMM	HMM	trajectory	trajectory
Decoding	HMM	trajectory	HMM	trajectory
Error rate (%)	19.7	19.5 (a)	72.1	18.0 (a)
rel. imp. (%)	ref	1.0	−266.0	8.6
Error rate (%)	15.3 (b)			9.0 (b)
rel. imp. (%)	ref			41.1

A lot of techniques for modeling of the inverse covariance (precision) matrices to capture intra-frame correlation efficiently in large vocaburary continuous speech recognition system have been proposed. Models that have been successfully applied include Semi-Tied Covariance matrices (STC) [28], [29], Extended Maximum Likelihood Linear Transform (EMLLT) [30], and Subspace for Precision And Mean (SPAM) [30]. The precision matrix models mentioned above can be described within a generic framework of basis superposition [31]. Proposed trajectory-HMM can be viewed as a basis superposition framework for temporal precision matrices. The $MT \times MT$ precision matrix $\boldsymbol{R}_{\boldsymbol{q}}$ is a weighted sum of DMT rank-1 symmetric matrices. This form is the same as an EMLLT for temporal precision matrix $\boldsymbol{R}_{\boldsymbol{q}}$. Both the basis and the diagonal matrices can be estimated in the general EMLLT framework. However, the basis matrix in the trajectory-HMM is a window coefficient matrix $\boldsymbol{W}$ given by Eq. (5). In addition, the diagonal matrix $\boldsymbol{\Sigma}_{\boldsymbol{q}}$ is given by transforming $\boldsymbol{\phi}$ by Gaussian distribution sequence matrix $\boldsymbol{S}_{\boldsymbol{q}}$ (Eq. (37)). Thus, temporal correlation can be captured efficiently without increasing the model parameters compared with the HMM. Furthermore, trajectory-HMM can be viewed as a SPAM model because mean vector is also constrained with in a subspace.

One interesting relationship between the basis superposition framework and Product-of-Gaussian (PoG) framework [32] has been shown [31]. The basis superposition is an example of a Product of Experts (PoE) system. Because the trajectory-HMM is an example of the basis superposition for the temporal precision matrix, it can also be viewed as the PoE system. Similar discussions can be found in [20], [21].

8. Conclusion

This paper described the definition and algorithms for the trajectory HMM. In the trajectory HMM framework, the standard HMM is naturally translated into a trajectory model by imposing the explicit relationship between static

(注4)：This algorithm can be viewed as a Kalman filtering for HMM mean sequence.

and dynamic features. We showed how it works as a trajectory model, and give an illustrative example with a new algorithm for sampling from the trajectory model. Although the parametrization of the trajectory model is the same as the standard HMM, it works quite differently. A speech recognition experiment demonstrated the importance of the consistency between training and decoding criteria. The relation to other techniques is also discussed.

Although,we used n-best rescoring strategy in the recognition experiment, it is expected that by decoding with the "trajectory likelihood," the recognition performance could be further improved. Therefore, our future work includes the implementation of a trajectory-based decoder for LVCSR.

Acknowledgment

Authors would like to thank Prof. Takao Kobayashi, Dr. Takashi Masuko, Dr. Yoshihiko Nankaku, and Dr. John Bridle for useful discussions.

References

[1] M. J. F. Gales and S. J. Young, "Segmental Hidden Markov Models," Proc. EUROSPEECH, pp.1579–1582, 1993.

[2] H. Gish and K. Ng, "Parametric trajectory models for speech recognition," Proc. ICSLP, pp.I-466–I-469, 1996.

[3] W. J. Holmes and M. J. Russell, "Speech recognition using a linear dynamic segmental HMM," Proc. ICASSP, pp.1611–1614, 1995.

[4] M. Ostendorf, V. Digalakis, and O. A. Kimball, "From HMMs to segment models: a unified view of stochastic modeling for speech recognition," IEEE Trans. on Speech and Audio Processing, vol.4, no.5, pp.360–378, 1996.

[5] L. Deng, M. Aksmanovic, X. Sun, J. Wu, "Speech recgnition using hiden Markov models with polynomial regression functions as nonstationary states," IEEE Trans. on Speech and Audio Processing, vol.2, no.4, pp.507–520, Oct. 1994.

[6] Y. Gong and J. P. Haton, "Stochastic trajectory modeling for speech recognition," Proc. ICASSP, vol.I, pp57–60, 1994.

[7] S. Takahashi, "Phoneme HMM's constrained by frame correlations," Proc. ICASSP, pp. 219-222, 1993.

[8] K. K. Paliwal, "Use of temporal correlation between successive frames in hidden Markov model based Speech recognizer," Proc. ICASSP, pp.215-218, 1993.

[9] M. Ostendorf and S.Roukos, "A stochastic segment model for phoneme-based continuous speech recognition," IEEE Trans. On Acoustics, Speech and Signal Processing, vol.37, no.12, pp.1857–1869, 1989.

[10] C. J. Wellekens, "Explicit correlation in hidden Markov model for Speech Recognition," Proc. ICASSP, pp.383-386, 1987.

[11] S. Furui, "Speaker independent isolated word rcognition using dynamic features of speech spectrum," IEEE Trans. Acoust., Speech, Signal Processing, vol.ASSP-34, pp.52–59, 1986.

[12] K. Tokuda, H. Zen, T. Kitamura, "A trajectory model derived from HMMs imposing explicit relationship between static and dynamic features," Proc. EUROSPEECH, pp.3189-3192, Sep. 2003.

[13] H. Zen, K. Tokuda, T. Kitamura "A Viterbi algorithm for a trajectory model derived from HMM with explicit relationship between static and dynamic features," Proc. of ICASSP, May 2004.

[14] H. Zen, K. Tokuda, T. Kitamura "An Introduction of trajectory model into HMM-based speech synthesis," Proc. of 5th ISCA Speech Synthesis Workshop, June 2004.

[15] K. Tokuda, T. Kobayashi and S. Imai, "Speech parameter generation from HMM using dynamic features," Proc. ICASSP, pp.660–663, 1995.

[16] K. Tokuda, T. Masuko, T. Yamada, T. Kobayashi and S. Imai, "An algorithm for speech parameter generation from continuous mixture HMMs with dynamic features," Proc. EUROSPEECH, pp.757–760, 1995.

[17] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," Proc. ICASSP, vol.3, pp.1315–1318, June 2000.

[18] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," Proc. EUROSPEECH, pp.2347–2350, 1999.

[19] Y. Minami, E. McDermott, A. Nakamura, S. Katagiri, "A recognition method with parametric trajectory synthesized using direct relations between static and dynamic feature vector time series," Proc. ICASSP, vol.I, pp.957–960, 2002.

[20] J. S. Bridle, "Towards Better Understanding of the Model Implied by the Use of Dynamic Features in HMMs," Proc. ICSLP, vol.1, pp.I-725–I-728, Oct. 2004.

[21] C. K. I. Williams, "How to pretend that correlated variables are independent by using difference observations," Neural Computation, accepted for publication.

[22] M. Tamura, T. Masuko, K. Tokuda and T. Kobayashi, "Speaker adaptation for HMM-based speech synthesis system using MLLR," in *Proc. ESCA/COCOSDA Third International Workshop on Speech Synthesis*, 1998, pp.273–276.

[23] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Speaker Interpolation in HMM-Based Speech Synthesis System," Proc. EUROSPEECH, pp.2523–2526, 1997.

[24] K. Shichiri, A. Sawabe, K. Tokuda, T. Masuko, T. Kobayashi, T. Kitamura, "Eigenvoices for HMM-based speech synthesis," Proc. ICSLP, pp.1269–1272, 2002.

[25] A. Acero, "Formant analysis and synthesis using hidden Markov models," Proc. EUROSPEECH, pp.1047–1050, 1999.

[26] K. Koishida, K. Tokuda, T. Masuko and T. Kobayashi, "Vector quantization of speech spectral parameters using statistics of dynamic features," Proc. ICSP, pp.247–252, 1997.

[27] Y. Minami, E. McDermott, A. Nakamura, A. Katagiri, "Recognition method with parametric trajectory generated from mixture distribution HMMs," Proc. of ICASSP, Vol. 1. pp. 124–127,2003.

[28] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," Computer Speech & Language, vol.12, no.2, pp.75–98, 1998.

[29] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov models," IEEE Transactions on Speech and Audio Processing, vol.7, no.3, pp.272–281, 1999.

[30] P. Olsen, R. Gopinath, "Modeling inverse covariance matrices by basis expansion," IEEE Transactions on Acoustic, Speech, and Signal Processing, vol.12, 37–46, 2004.

[31] K. Sim, M. Gales, "Precision matrix modeling for large vocabulary continuous speech recognition," Tech. Rep. CUED/F-INFENG/TR.485, Cambridge University, 2004.

[32] M. Gales, S. Airey, "Product of Gaussians for speech recognition," Tech. Rep. CUED/F-INFENG/TR.458, Cambridge University, 2003.