Bayesian Speech Synthesis Framework Integrating Training and Synthesis Processes

Kei Hashimoto, Yoshihiko Nankaku, Keiichi Tokuda

Department of Scientific and Engineering Simulation, Nagoya Institute of Technology, Nagoya, JAPAN

Abstract

This paper proposes a speech synthesis technique integrating training and synthesis processes based on the Bayesian framework. In the Bayesian speech synthesis, all processes are derived from one single predictive distribution which represents the problem of speech synthesis directly. However, it typically assumes that the posterior distribution of model parameters is independent of synthesis data, and this separates the system into training and synthesis parts. This paper removes the approximation and derives an algorithm that the posterior distributions, decision trees and synthesis data are iteratively updated. Experimental results show that the proposed method improves the quality of synthesized speech.

Index Terms: speech synthesis, HMM, Bayesian approach

1. Introduction

A statistical speech synthesis system based on hidden Markov models (HMMs) was recently developed. In HMM-based speech synthesis, the spectrum, excitation and duration of speech are simultaneously modeled with HMMs, and speech parameter sequences are generated from the HMMs themselves [1]. The maximum likelihood (ML) criterion has typically been used for training HMMs and generating speech parameters. The ML criterion guarantees that the ML estimates approach the true values of the parameters. Therefore, acoustic modeling based on HMMs have been developed greatly by using the ML approach. However, since the ML criterion produces a point estimate of the HMM parameters, its estimation accuracy may deteriorate when the amount of training data is insufficient.

The Bayesian approach considers the posterior distribution of variables. That is, all variables introduced when the models are parameterized, such as the model parameters and latent variables, are treated as random variables, and their posterior distributions are obtained by invoking the Bayes theorem. The difference between the Bayesian and ML approaches is that the target of estimation is the distribution function in the Bayesian approach whereas it is the parameter value in the ML approach. Because of its posterior distribution estimation, the Bayesian approach can generally construct a more robust model than the ML approach can. However, the Bayesian approach requires complicated integral and expectation computations to obtain posterior distributions. It is difficult to solve these computations without approximations when the models have latent variables. To avoid complicated computations, a variational Bayes (VB) method has been proposed in the field of learning theory [4]. This method can obtain approximate posterior distributions through iterative calculations similar to the expectation-maximization (EM) algorithm used in the ML approach.

Recently, a Bayesian framework to HMM-based speech synthesis has been proposed [2, 3]. We call this framework Bayesian speech synthesis. In Bayesian speech synthesis, all processes for constructing the system can be derived from one single predictive distribution that directly represents the problem of speech synthesis. The estimation of the posterior distributions, model selection, and speech parameter generation are consistently performed by maximizing the log marginal likelihood. The posterior distributions of all variables are obtained by using the VB method. Then, the obtained posterior distribution of the model parameters depends on not only the training data, but also the synthesis data. In a basic speech synthesis situation, the observed data for the synthesis sentences is not given beforehand. Therefore, the posterior distributions cannot be obtained. To overcome this problem, it typically assumes that the posterior distribution of the model parameters is independent of the synthesis data [2, 3]. As a result of this approximation, the Bayesian speech synthesis system is separated into training and synthesis parts, as the conventional ML-based system, and the posterior distribution of the model parameters and decision trees can be obtained from only the training data. However, although the posterior distributions can be estimated, they don't consider synthesis data, and the system doesn't represent the Bayesian speech synthesis exactly. This paper proposes a speech synthesis technique integrating training and synthesis processes based on the Bayesian framework. This method removes the approximation and leads to an algorithm that the posterior distributions, decision trees and synthesis data are iteratively updated.

The rest of this paper is organized as follows. Section 2 describes Bayesian speech synthesis. Section 3 proposes the Bayesian speech synthesis framework integrating the training and synthesis processes. Subjective listening test results are presented in Section 4. Concluding remarks and future work are presented in the final section.

2. Bayesian speech synthesis

2.1. Bayesian approach

The output distribution is obtained from the left-to-right HMM that has been widely used to represent acoustic models for speech synthesis. Let $\boldsymbol{O} = (\boldsymbol{O}_1, \boldsymbol{O}_2, \ldots, \boldsymbol{O}_T)$ be a set of training data of D dimensional feature vectors, and let T denote the frame number. The output distribution is represented by

$$\begin{aligned}\log P(\boldsymbol{O}, \boldsymbol{Z} \mid \boldsymbol{\Lambda}) &= \sum_{i=1}^{N} Z_1^i \log \pi_i + \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} Z_t^i Z_{t+1}^j \log a_{ij} \\ &\quad + \sum_{t=1}^{T} \sum_{i=1}^{N} Z_t^i \log \mathcal{N}(\boldsymbol{O}_t \mid \boldsymbol{\mu}_i, \boldsymbol{S}_i^{-1}) \end{aligned} \quad (1)$$

where $\boldsymbol{Z} = (Z_1, Z_2, \ldots, Z_T)$ is a sequence of latent variables which represent HMM states, $Z_t \in \{1, \ldots, N\}$ denotes a state at frame t, and N is the number of states in an HMM.

$$Z_t^i = \delta(Z_t, i) = \begin{cases} 1 & \text{if } Z_t = i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The set of model parameters $\boldsymbol{\Lambda} = \{\pi_i, a_{ij}, \boldsymbol{\mu}_i, \boldsymbol{S}_i\}_{i,j=1}^N$ consists of the initial state probability π_i of state i, the state transition probability a_{ij} from state i to state j, the mean vector $\boldsymbol{\mu}_i$, and the covariance matrix $\boldsymbol{S}_i^{-1}$ of a Gaussian distribution $\mathcal{N}(\cdot \mid \boldsymbol{\mu}_i, \boldsymbol{S}_i^{-1})$.

In HMM-based speech synthesis, the ML criterion has typically been used to train HMMs and generate speech parameters. The optimal model parameters can be obtained by maximizing the likelihood for given training data.

$$\boldsymbol{\Lambda}_{ML} = \arg\max_{\boldsymbol{\Lambda}} P(\boldsymbol{O} \mid S, \boldsymbol{\Lambda}) \tag{3}$$

where S is a label sequence of training data. Since it is difficult to obtain the model parameter $\boldsymbol{\Lambda}_{ML}$ analytically, the model parameters are estimated by using an iterative procedure such as the EM algorithm. In the synthesis part, the speech parameter generation algorithm generates sequences of speech parameter vectors that maximize their output probabilities by using the model parameters $\boldsymbol{\Lambda}_{ML}$.

$$\hat{\boldsymbol{o}}_{ML} = \arg\max_{\boldsymbol{o}} P(\boldsymbol{o} \mid s, \boldsymbol{\Lambda}_{ML}) \tag{4}$$

where $\boldsymbol{o} = \left[\boldsymbol{o}_1^\top, \boldsymbol{o}_2^\top, \ldots, \boldsymbol{o}_T^\top\right]^\top$ is a speech parameter sequence, and s is a label sequence to be synthesized. The ML criterion guarantees that the ML estimates approach the true values of the parameters. However, since the ML criterion produces a point estimate of the HMM parameters, the estimation accuracy may deteriorate when the amount of training data is insufficient.

The Bayesian approach assumes that a set of model parameters $\boldsymbol{\Lambda}$ is a random variable, while the ML approach estimates constant model parameters. In the Bayesian approach, the speech parameter is generated from a predictive distribution as follows.

$$\begin{aligned} \hat{\boldsymbol{o}}_{Bayes} &= \arg\max_{\boldsymbol{o}} P(\boldsymbol{o} \mid s, \boldsymbol{O}, S) \\ &= \arg\max_{\boldsymbol{o}} P(\boldsymbol{o}, \boldsymbol{O} \mid s, S) \end{aligned} \tag{5}$$

It can be seen that Eq. (5) directly represents the problem of speech synthesis; that is, the speech feature sequence $\boldsymbol{o}$ is generated from given training feature sequences $\boldsymbol{O}$ with labels S and labels to be synthesized s. The marginal likelihood of $\boldsymbol{o}$ and $\boldsymbol{O}$ is defined by

$$\begin{aligned} &P(\boldsymbol{o}, \boldsymbol{O} \mid s, S) \\ &= \sum_{\boldsymbol{z}} \sum_{\boldsymbol{Z}} \int P(\boldsymbol{o}, \boldsymbol{z}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S) d\boldsymbol{\Lambda} \\ &= \sum_{\boldsymbol{z}} \sum_{\boldsymbol{Z}} \int P(\boldsymbol{o}, \boldsymbol{z} \mid s, \boldsymbol{\Lambda}) P(\boldsymbol{O}, \boldsymbol{Z} \mid S, \boldsymbol{\Lambda}) P(\boldsymbol{\Lambda}) d\boldsymbol{\Lambda} \end{aligned} \tag{6}$$

where $\boldsymbol{z}$ is a sequence of HMM states for a speech parameter sequence $\boldsymbol{o}$, $P(\boldsymbol{\Lambda})$ is the prior distribution for model parameter $\boldsymbol{\Lambda}$, $P(\boldsymbol{o}, \boldsymbol{z} \mid s, \boldsymbol{\Lambda})$ is the likelihood of synthesis data $\boldsymbol{o}$, and $P(\boldsymbol{O}, \boldsymbol{Z} \mid S, \boldsymbol{\Lambda})$ is the likelihood of training data $\boldsymbol{O}$. The model parameters are integrated out in Eq. (6) so that the effect of over-fitting is mitigated. However, it is difficult to solve the integral and expectation calculations. The calculations become more complicated when a model includes latent variables. The variational Bayesian method has been proposed as a tractable approximation to overcome this problem, and it has good generalization performance in many applications [4].

2.2. Variational Bayesian method

The variational Bayesian method maximizes the lower bound of the log marginal likelihood $\mathcal{F}$ instead of the true marginal likelihood. The lower bound $\mathcal{F}$ is defined by using Jensen's inequality:

$$\begin{aligned} &\log P(\boldsymbol{o}, \boldsymbol{O} \mid s, S) \\ &= \log \sum_{\boldsymbol{z}} \sum_{\boldsymbol{Z}} \int P(\boldsymbol{o}, \boldsymbol{z}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S) d\boldsymbol{\Lambda} \\ &= \log \sum_{\boldsymbol{z}} \sum_{\boldsymbol{Z}} \int Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda}) \frac{P(\boldsymbol{o}, \boldsymbol{z}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S)}{Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda})} d\boldsymbol{\Lambda} \\ &\geq \sum_{\boldsymbol{z}} \sum_{\boldsymbol{Z}} \int Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda}) \log \frac{P(\boldsymbol{o}, \boldsymbol{z}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S)}{Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda})} d\boldsymbol{\Lambda} \\ &= \left\langle \log \frac{P(\boldsymbol{o}, \boldsymbol{z}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S)}{Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda})} \right\rangle_{Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda})} \\ &= \mathcal{F} \end{aligned} \tag{7}$$

where $\langle \cdot \rangle_Q$ denotes a calculation of the expectation with respect to Q, and $Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda})$ is an approximate distribution of the true posterior distribution $P(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid \boldsymbol{o}, \boldsymbol{O}, s, S)$. The VB method assumes that the probabilistic variables associated with $\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda}$ are statistically independent of the other variables.

$$Q(\boldsymbol{z}, \boldsymbol{Z}, \boldsymbol{\Lambda}) = Q(\boldsymbol{z})\, Q(\boldsymbol{Z})\, Q(\boldsymbol{\Lambda}) \tag{8}$$

In the VB method, posterior distributions $Q(\boldsymbol{z})$, $Q(\boldsymbol{Z})$ and $Q(\boldsymbol{\Lambda})$ are introduced to approximate the true posterior distributions. The optimal posterior distributions can be obtained by maximizing the objective function $\mathcal{F}$ with the variational method.

$$Q(\boldsymbol{z}) = C_{\boldsymbol{z}} \exp \langle \log P(\boldsymbol{o}, \boldsymbol{z} \mid s, \boldsymbol{\Lambda}) \rangle_{Q(\boldsymbol{\Lambda})} \tag{9}$$

$$Q(\boldsymbol{Z}) = C_{\boldsymbol{Z}} \exp \langle \log P(\boldsymbol{O}, \boldsymbol{Z} \mid S, \boldsymbol{\Lambda}) \rangle_{Q(\boldsymbol{\Lambda})} \tag{10}$$

$$\begin{aligned} Q(\boldsymbol{\Lambda}) = C_{\boldsymbol{\Lambda}} P(\boldsymbol{\Lambda}) &\exp \langle \log P(\boldsymbol{o}, \boldsymbol{z} \mid s, \boldsymbol{\Lambda}) \rangle_{Q(\boldsymbol{z})} \\ &\times \exp \langle \log P(\boldsymbol{O}, \boldsymbol{Z} \mid S, \boldsymbol{\Lambda}) \rangle_{Q(\boldsymbol{Z})} \end{aligned} \tag{11}$$

where $C_{\boldsymbol{z}}$, $C_{\boldsymbol{Z}}$ and $C_{\boldsymbol{\Lambda}}$ are normalization terms of $Q(\boldsymbol{z})$, $Q(\boldsymbol{Z})$ and $Q(\boldsymbol{\Lambda})$, respectively. These posterior distributions can be updated by using iterative calculations similar to those of the EM algorithm in the ML approach.

2.3. Bayesian model selection

According to the Bayes theorem, the posterior distribution of a model structure can be represented by

$$P(m \mid \boldsymbol{o}, \boldsymbol{O}) = \frac{P(\boldsymbol{o}, \boldsymbol{O} \mid m) P(m)}{P(\boldsymbol{o}, \boldsymbol{O})} \tag{12}$$

If the optimal model structure $\hat{m}$ is selected by maximizing the posterior probability, the optimal model structure can be obtained from Eq. (12).

$$\begin{aligned} \hat{m} &= \arg\max_m \frac{P(\boldsymbol{o}, \boldsymbol{O} \mid m) P(m)}{P(\boldsymbol{o}, \boldsymbol{O})} \\ &= \arg\max_m P(\boldsymbol{o}, \boldsymbol{O} \mid m) P(m) \end{aligned} \tag{13}$$

By applying the VB method and using the assumption that the prior distribution $P(m)$ is a uniform distribution, the optimal model structure $\hat{m}$ can be determined as follows:

$$\begin{aligned}\hat{m} &= \arg\max_{m} \log P(\boldsymbol{o}, \boldsymbol{O} \mid m) \\ &\approx \arg\max_{m} \mathcal{F} \end{aligned} \tag{14}$$

Consequently, an optimal model structure can be selected by maximizing the objective function $\mathcal{F}$ [5, 6].

2.3.1. Bayesian context clustering

The decision tree based context clustering [7, 8] is a top-down clustering method to optimize the state tying structure for a robust model parameter estimation. A leaf of the decision tree corresponds to a set of HMM states to be tied. The decision tree growing process begins with the root node, which has all HMM states to be tied. Then, a question which divides the set of states into two subsets assigned respectively to two child nodes, the "Yes" node and the "No" node, is chosen so as to maximize the value of an objective function. The decision tree is grown in greedy fashion by splitting node that maximizes the gain of the objective function at each step. In the HMM-based speech synthesis, model parameters of the spectrum, excitation, and duration are separately clustered because they have their own contextual factors.

When a node is split into a "Yes" node and a "No" node by question q, the gain $\Delta\mathcal{F}_q$ is defined as the difference of $\mathcal{F}$ before and after splitting:

$$\Delta\mathcal{F}_q = \mathcal{F}_q^y + \mathcal{F}_q^n - \mathcal{F}_q^p \tag{15}$$

where $\mathcal{F}_q^y$ and $\mathcal{F}_q^n$ are the values of the objective function $\mathcal{F}$ of the nodes split by question q, and $\mathcal{F}_q^p$ is the value before splitting. The question $\hat{q}$ is chosen from the question set as follows:

$$\hat{q} = \arg\max_{q} \Delta\mathcal{F}_q \tag{16}$$

The decision tree that maximizes the objective function $\mathcal{F}$ is obtained by splitting nodes until $\Delta\mathcal{F}_{\hat{q}} \leq 0$.

2.3.2. Bayesian context clustering using cross validation

The prior distributions are heuristically determined in many cases, because the prior data is not usually given in HMM-based speech synthesis. However, hyper-parameters (the parameters of the prior distributions) affect the model selection as tuning parameters. Therefore, a determination technique of prior distribution is required to automatically select an appropriate model structure. One possible approach is to optimize the hyper-parameters by using training data so as to maximize the marginal likelihood. However, this approach still needs tuning parameters to control the influences of prior distributions, and it often leads to the over-fitting problem as in the case of the ML criterion. To overcome this problem, the prior distribution determination technique using cross validation has been proposed [9]. Here, we apply it to context clustering.

Let $\boldsymbol{O} = \{\boldsymbol{O}^{(1)}, \boldsymbol{O}^{(2)}, \ldots, \boldsymbol{O}^{(k)}, \ldots, \boldsymbol{O}^{(K)}\}$ be the set of training data and $\boldsymbol{O}^{(k)}$ be a partition for K-fold cross validation. For the k-th evaluation, $\boldsymbol{O}^{(\bar{k})} = \{\boldsymbol{O}^{(j)} \mid j \neq k\}$ is used to determine the prior distributions and $\boldsymbol{O}^{(k)}$ is used to estimate the posterior distributions. Accordingly, the Bayesian approach using cross validation calculates the log marginal likelihood $\log P(\boldsymbol{O}^{(k)} \mid \boldsymbol{O}^{(\bar{k})}, S)$. Using Jensen's inequality, the lower bound of log marginal likelihood $\mathcal{F}^{(k)}$ can be defined as Eq. (7):

$$\log P(\boldsymbol{O}^{(k)} \mid \boldsymbol{O}^{(\bar{k})}, s, S) \geq \mathcal{F}^{(k)} \tag{17}$$

For the k-th evaluation, the optimal VB posterior distributions of the model parameters can be obtained by using the variational method to maximize $\mathcal{F}^{(k)}$ with respect to $Q(\boldsymbol{\Lambda}^{(k)})$:

$$\begin{aligned}Q(\boldsymbol{\Lambda}^{(k)}) =& C_{\boldsymbol{\Lambda}^{(k)}} P(\boldsymbol{\Lambda}^{(k)} \mid \boldsymbol{O}^{(\bar{k})}) \\ & \times \left\langle \log P(\boldsymbol{O}^{(k)}, \boldsymbol{Q}^{(k)} \mid \boldsymbol{\Lambda}^{(k)}) \right\rangle_{Q(\boldsymbol{Q}^{(k)})}\end{aligned} \tag{18}$$

where $P(\boldsymbol{\Lambda}^{(k)} \mid \boldsymbol{O}^{(\bar{k})})$ is a prior distribution that represents the prior information $\boldsymbol{O}^{(\bar{k})}$ and $C_{\boldsymbol{\Lambda}^{(k)}}$ is a normalization term.

The objective function of the Bayesian approach using cross validation $\mathcal{F}^{(CV)}$ is obtained by summing $\mathcal{F}^{(k)}$ for each fold:

$$\mathcal{F}^{(CV)} = \sum_{k=1}^{K} \mathcal{F}^{(k)} \tag{19}$$

An optimal model structure can be selected by maximizing the objective function $\mathcal{F}^{(CV)}$ instead of $\mathcal{F}$. The question which maximizes the gain of the objective function $\Delta\mathcal{F}_q^{(CV)}$ is selected as in Eq. (16). The decision tree that maximizes the objective function $\mathcal{F}^{(CV)}$ is obtained by splitting nodes until $\Delta\mathcal{F}_{\hat{q}}^{(CV)} \leq 0$.

3. Bayesian speech synthesis integrating training and synthesis processes

3.1. Speech parameter generation

In the synthesis part of HMM-based speech synthesis, first, an arbitrarily given text to be synthesized is converted into a context-dependent label sequence and a sentence HMM is constructed by concatenating context-dependent HMMs according to the label sequence. Second, the optimal state sequence of the sentence HMM is determined. Third, a speech parameter sequence is generated for a given state sequence. From Eq. (5), the optimal speech parameter sequence for Bayesian speech synthesis can be generated by maximizing the marginal likelihood. Thus, the optimal speech parameter sequence $\hat{\boldsymbol{o}}$ can be generated by maximizing the lower bound $\mathcal{F}$ in Eq. (7) because the VB method guarantees that the log marginal likelihood is approximately the lower bound $\mathcal{F}$.

$$\begin{aligned}\hat{\boldsymbol{o}}_{Bayes} &= \arg\max_{\boldsymbol{o}} \log P(\boldsymbol{o}, \boldsymbol{O} \mid s, S) \\ &\approx \arg\max_{\boldsymbol{o}} \mathcal{F}\end{aligned} \tag{20}$$

We assume that a speech parameter vector $\boldsymbol{o}_t$ consists of a static feature vector $\boldsymbol{c}_t$ and its first and second order dynamic feature vectors.

$$\begin{aligned}\boldsymbol{o} &= \boldsymbol{W}\boldsymbol{c} \\ &= \left[(\boldsymbol{W}\boldsymbol{c})_1^\top, (\boldsymbol{W}\boldsymbol{c})_2^\top, \ldots, (\boldsymbol{W}\boldsymbol{c})_T^\top\right]^\top \end{aligned} \tag{21}$$

$$(\boldsymbol{W}\boldsymbol{c})_t = \left[\boldsymbol{c}_t^\top, \Delta\boldsymbol{c}_t^\top, \Delta^2\boldsymbol{c}_t^\top\right]^\top \tag{22}$$

where $\boldsymbol{W}$ is a window matrix to calculate dynamic features from static features [10]. The dynamic feature vectors are automatically determined from the window matrix $\boldsymbol{W}$ and the static

feature sequence. Consequently, only a static feature vector sequence $\boldsymbol{c}$ is estimated in the synthesis part. From Eq. (20), the optimal static feature sequence $\hat{\boldsymbol{c}}$ is generated by maximizing the lower bound $\mathcal{F}$. Moreover, under the condition of Eq. (21), the optimal static feature sequence $\hat{\boldsymbol{c}}$ can be determined by solving the following equation:

$$\frac{\partial \mathcal{F}}{\partial \boldsymbol{c}} = \frac{\partial}{\partial \boldsymbol{c}} \left\langle \log \frac{P(\boldsymbol{W}\boldsymbol{c}, \boldsymbol{z}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S)}{Q(\boldsymbol{z})Q(\boldsymbol{Z})Q(\boldsymbol{\Lambda})} \right\rangle_{Q(\boldsymbol{z})Q(\boldsymbol{Z})Q(\boldsymbol{\Lambda})} = \boldsymbol{0} \tag{23}$$

In the Bayesian speech synthesis framework, the estimation of the posterior distributions, model selection, and speech parameter generation consistently maximize the lower bound $\mathcal{F}$.

3.2. Approximation for estimating posterior distributions

The obtained posterior distribution of model parameters $Q(\boldsymbol{\Lambda})$ in Eq. (11) depend on not only the training data $\boldsymbol{O}$, but also the synthesis data $\boldsymbol{o}$. However, in a basic speech synthesis situation, the observed data of synthesis sentences is not given previously. Therefore, the posterior distributions represented in Eq. (11) cannot be estimated. To overcome this problem, one typically assumes that the posterior distribution of the model parameters is independent of the synthesis data [2, 3]. The lower bound of the log marginal likelihood with respect to only the training data $\boldsymbol{O}$ can be represented as follows.

$$\begin{aligned} \log P(\boldsymbol{O} \mid S) &= \log \sum_{\boldsymbol{Z}} \int P(\boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid S) d\boldsymbol{\Lambda} \\ &\geq \left\langle \log \frac{P(\boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid S)}{\bar{Q}(\boldsymbol{Z})\bar{Q}(\boldsymbol{\Lambda})} \right\rangle_{\bar{Q}(\boldsymbol{Z})\bar{Q}(\boldsymbol{\Lambda})} \\ &= \bar{\mathcal{F}} \end{aligned} \tag{24}$$

The posterior distributions $\bar{Q}(\boldsymbol{Z})$ and $\bar{Q}(\boldsymbol{\Lambda})$ can be estimated by maximizing the lower bound $\bar{\mathcal{F}}$. The posterior distribution of the model parameters $\bar{Q}(\boldsymbol{\Lambda})$ is represented as follows.

$$\bar{Q}(\boldsymbol{\Lambda}) = \bar{C}_{\boldsymbol{\Lambda}} P(\boldsymbol{\Lambda}) \exp \langle \log P(\boldsymbol{O}, \boldsymbol{Z} \mid S, \boldsymbol{\Lambda}) \rangle_{\bar{Q}(\boldsymbol{Z})} \tag{25}$$

Equation (25) indicates that the posterior distribution $\bar{Q}(\boldsymbol{\Lambda})$ is independent of the synthesis data and that it can be estimated by using only the training data. Since the same approximation is used in the Bayesian model selection, the optimal decision trees are selected by maximizing the lower bound $\bar{\mathcal{F}}$ instead of $\mathcal{F}$.

$$\hat{m} = \arg\max_{m} \bar{\mathcal{F}} \tag{26}$$

Consequently, the decision trees are selected independently of the synthesis data. Additionally, Eq. (23) can be represented by the estimated posterior distribution $\bar{Q}(\boldsymbol{\Lambda})$ and the determined state sequence as follows.

$$\left\langle \frac{\partial}{\partial \boldsymbol{c}} \log P(\boldsymbol{W}\boldsymbol{c} \mid \boldsymbol{z}, \boldsymbol{\Lambda}) \right\rangle_{\bar{Q}(\boldsymbol{\Lambda})} = \boldsymbol{0} \tag{27}$$

Equation (27) can be solved efficiently by using the Cholesky or QR decomposition [10]. Therefore, the computational cost is almost the same as the ML criterion.

The approximation that the posterior distribution of the model parameters is independent of the synthesis data $\boldsymbol{o}$ enables the Bayesian speech synthesis system to be separated into training and synthesis parts as the conventional ML-based system and to obtain the posterior distribution of model parameters and decision trees from only the training data. However, although the posterior distributions can be estimated, they don't take into account synthesis data, and the system doesn't represent the Bayesian speech synthesis exactly. To overcome this problem, this paper proposes a speech synthesis technique integrating training and synthesis processes based on the Bayesian framework.

3.3. Integration of training and synthesis processes

The proposed method removes the approximation and derives an algorithm that the posterior distributions, decision trees, and synthesis data are iteratively updated. In the proposed framework, the generated speech parameters of the synthesis sentences are used instead of the observed data. That is, the posterior distributions and decision trees are estimated from the training data and the generated speech parameters, and the speech parameters are generated from the estimated posterior distributions. Since the posterior distributions, decision trees, and generated speech parameters depend on each other, they are iteratively updated as the EM algorithm. Initial synthesis data are generated by using the framework described in the preceding section 3.2. Once the generated speech parameters are obtained, they can be used for estimating the posterior distribution. The new lower bound with the generated speech parameters is defined as follows.

$$\begin{aligned} &\log P(\tilde{\boldsymbol{o}}, \boldsymbol{O} \mid s, S) \\ &\quad = \log \sum_{\tilde{\boldsymbol{z}}} \sum_{\boldsymbol{Z}} \int P(\tilde{\boldsymbol{o}}, \tilde{\boldsymbol{z}}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S) d\boldsymbol{\Lambda} \\ &\quad \geq \left\langle \log \frac{P(\tilde{\boldsymbol{o}}, \tilde{\boldsymbol{z}}, \boldsymbol{O}, \boldsymbol{Z}, \boldsymbol{\Lambda} \mid s, S)}{\tilde{Q}(\tilde{\boldsymbol{z}})\tilde{Q}(\boldsymbol{Z})\tilde{Q}(\boldsymbol{\Lambda})} \right\rangle_{\tilde{Q}(\tilde{\boldsymbol{z}})\tilde{Q}(\boldsymbol{Z})\tilde{Q}(\boldsymbol{\Lambda})} \\ &\quad = \tilde{\mathcal{F}} \end{aligned} \tag{28}$$

where $\tilde{\boldsymbol{o}}$ is the generated speech parameter sequence. By maximizing the lower bound $\tilde{\mathcal{F}}$, the posterior distribution can be estimated in the same fashion as Eq. (11).

$$\begin{aligned} \tilde{Q}(\boldsymbol{\Lambda}) = \tilde{C}_{\boldsymbol{\Lambda}} P(\boldsymbol{\Lambda}) &\exp \langle \log P(\tilde{\boldsymbol{o}}, \tilde{\boldsymbol{z}} \mid s, \boldsymbol{\Lambda}) \rangle_{\tilde{Q}(\tilde{\boldsymbol{z}})} \\ &\times \exp \langle \log P(\boldsymbol{O}, \boldsymbol{Z} \mid S, \boldsymbol{\Lambda}) \rangle_{\tilde{Q}(\boldsymbol{Z})} \end{aligned} \tag{29}$$

The posterior distributions are estimated from the training data and the generated speech parameters instead of the observed speech parameters. Additionally, the decision trees are selected by maximizing the lower bound $\tilde{\mathcal{F}}$.

$$\hat{m} = \arg\max_{m} \tilde{\mathcal{F}} \tag{30}$$

Equation (23) can be represented by the estimated posterior distribution $\tilde{Q}(\boldsymbol{\Lambda})$ and the determined state sequence.

$$\left\langle \frac{\partial}{\partial \boldsymbol{c}} \log P(\boldsymbol{W}\boldsymbol{c} \mid \boldsymbol{z}, \boldsymbol{\Lambda}) \right\rangle_{\tilde{Q}(\boldsymbol{\Lambda})} = \boldsymbol{0} \tag{31}$$

In the proposed framework, the estimation of posterior distributions, model selection and speech parameter generation consistently maximize the lower bound $\tilde{\mathcal{F}}$. The posterior distributions, decision trees, and synthesis data are iteratively updated. The iterative process is as follows.

1. Initial speech parameters of synthesis sentences are generated with in the represented framework (Eq. (27)).
2. The posterior distributions $\tilde{Q}(\tilde{\boldsymbol{z}})\tilde{Q}(\boldsymbol{Z})\tilde{Q}(\boldsymbol{\Lambda})$ and decision trees are re-estimated by maximizing the lower bound $\tilde{\mathcal{F}}$ (Eqs. (29) and (30)).

3. Speech parameters of synthesis sentences are re-generated by using the estimated posterior distribution (Eq. (31)).

4. Steps 2 and 3 are iterated until the value of $\tilde{\mathcal{F}}$ converge.

Although the iterative process increase the computational cost, the final posterior distributions is more appropriate than one used in the previous method for synthesis sentences.

The key question is *how many synthesis sentences should be used for estimating the posterior distributions?* Here, we discuss two approaches about the number of synthesis sentences.

- **Sentence:** The generated speech parameters of one synthesis sentence are used as $\tilde{o}$.
- **Batch:** The generated speech parameters of all synthesis sentences are used as $\tilde{o}$.

Sentence estimates different posterior distributions and model structures for each synthesis sentence. On the other hand, **Batch** estimates the same posterior distributions and model structures for all synthesis sentences. Therefore, **Sentence** needs the larger computational cost than **Batch**.

4. Experiments

4.1. Experimental conditions

The experiments used the ATR Japanese speech database [11] B-set, which consists of 503 phonetically balanced sentences. The first 450 of the 503 sentences, uttered by one male speaker (MHT), were used for training. The remaining 53 sentences were used for the evaluations. Speech signals were sampled at a rate of 16 kHz and windowed at a 5-ms frame rate using a 25-ms Blackman window. Feature vectors consisted of spectrum and F_0 parameter vectors. The spectrum parameter vectors consisted of 24 mel-cepstral coefficients, and their delta and delta-delta coefficients. The F_0 parameter vectors consisted of log F_0 and its delta and delta-delta. A five-state, left-to-right MSD-HSMM [12, 13] without skip transitions was used. Each state output PDF was composed of spectrum and F_0 streams. The spectrum stream was modeled by single multi-variate Gaussian distributions with diagonal covariance matrices. The F_0 stream was modeled by a multi-space probability distribution consisting of a Gaussian distribution for voiced frames and a discrete distribution for unvoiced frames. Each state duration PDF was modeled by a one-dimensional Gaussian distribution. The decision tree-based context clustering technique was separately applied to distributions of spectrum, F_0, and state duration.

A subjective listening test was conducted to evaluate the quality of the synthesized speech. The test assessed the naturalness of the converted speech by the mean opinion score (MOS) test method. The subjects were 10 Japanese students belonging to our research group. Twenty sentences were chosen at random from the evaluation sentences. Samples were presented in random order for each synthesis sentence. In the MOS test, after listening to each test sample, the subjects were asked to assign the sample a five-point naturalness score (5: natural – 1: poor).

4.2. Comparing the number of updates

This experiment evaluated the effectiveness of the proposed iterative updates by comparing the following four systems.

- **Iteration0** : The posterior distributions were trained from only the training data.

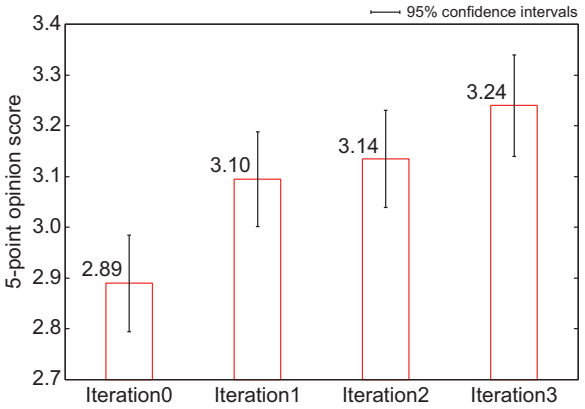

Figure 1: Mean opinion scores of speech synthesized by the baseline and proposed methods. Error bars show 95% confidence intervals.

- **Iteration1** : The posterior distributions were trained from the training data and the speech parameters generated by **Iteration0**.
- **Iteration2** : The posterior distributions were trained from the training data and the speech parameters generated by **Iteration1**.
- **Iteration3** : The posterior distributions were trained from the training data and the speech parameters generated by **Iteration2**.

Iteration0 was the baseline Bayesian speech synthesis system described in Section 3.2. **Iteration1**, **Iteration2**, and **Iteration3** were the proposed system integrating training and synthesis processes described in Section 3.3, and they were based on sentence-form integration. In each iteration, the posterior distributions were updated five times. Therefore, in this experiment, the number of updates was different for each system.

Figure 1 plots the experimental results. Although there were not confidence intervals, it is clear that the subjective score increased as the number of training iterations increased. These results clearly show the effectiveness of the training and synthesis iterations. The decision trees constructed in the context clustering varied between the four systems. This shows that the posterior distributions were optimized as a result of integrating the training and synthesis processes.

4.3. Comparing systems

This experiment compared the following four systems.

- **ML** : The conventional ML-based speech synthesis system. The HMMs were trained by using the ML criterion. The decision trees were selected by the MDL criterion [14].
- **Baseline** : The baseline Bayesian speech synthesis system described in Section 3.2.
- **Batch** : The proposed Bayesian speech synthesis system based on the batch-form integration described in Section 3.3.
- **Sentence** : The proposed Bayesian speech synthesis system based on the sentence-form integration described in Section 3.3. This system was the same as **Iteration3** of the previous experiment.

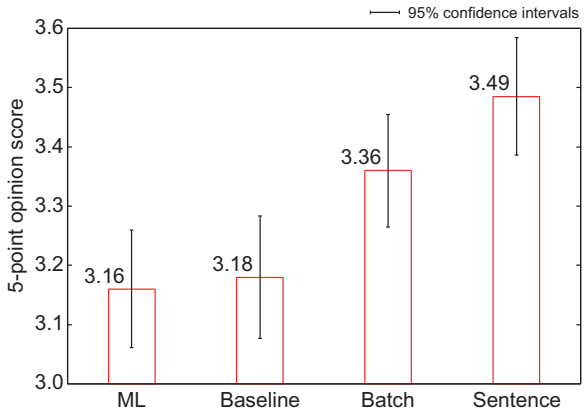

Figure 2: Mean opinion scores of speech synthesized by the baseline and proposed methods. Error bars show 95% confidence intervals.

The computational costs of **ML**, **Baseline**, and **Batch** were almost same because the number of updates was same in this experiment. However, since **Sentence** estimated different posterior distributions and model structures for each synthesis sentence, the computational cost was 53 times as large as **Batch**.

Figure 2 shows the results of the subjective listening test. **Baseline** was better than **ML**, although the gain was not significant. In addition, **Batch** and **Sentence** outperformed **Baseline**. These performance gains illustrate the effectiveness of the proposed Bayesian speech synthesis framework integrating training and synthesis processes. The figure also shows that **Sentence** performed better than **Batch**. Although **Batch** used all generated synthesis data to estimate the posterior distributions, the posterior distributions and model structures of **Batch** were common for all synthesis sentences. In contrast, **Sentence** estimated different posterior distributions and model structures for each synthesis sentence. The experimental results illustrate that the quality of the synthesized speech improved when the posterior distributions were optimized for each synthesis sentence.

5. Conclusions

This paper proposes a speech synthesis technique integrating training and synthesis processes based on the Bayesian framework. The proposed method removes the approximation that the posterior distribution of the model parameters is independent of the synthesis data and derives an algorithm that the posterior distributions, decision trees and synthesis data are iteratively updated. Both sentence-form and batch-form integrations were tested. The sentence-form integration estimates different posterior distributions and decision trees for each synthesis sentence, whereas the batch-form integration estimates the same ones for all synthesis sentences. The results of MOS synthesis demonstrated that the proposed method outperforms the baseline method and the sentence-form integration performed better than the batch-form integration.

Our future work will include investigation of the relation between the amount of training data and the quality of speech synthesized by the proposed method.

6. Acknowledgements

The research leading to these results was partly funded from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement 213845 (the EMIME project http://www.emime.org). A part of this research was supported by JSPS (Japan Society for the Promotion of Science) Research Fellowships for Young Scientists.

7. References

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

[2] K. Hashimoto, H. Zen, Y. Nankaku, T. Masuko, and K. Tokuda, "A Bayesian approach to HMM-based speech synthesis," in Proc. ICASSP, pp.4029–4032, 2009.

[3] K. Hashimoto, Y. Nankaku, and K. Tokuda, "A Bayesian approach to hidden semi Markov model based speech synthesis," in Proc. Interspeech 2009, pp.1751–1754, 2009.

[4] H. Attias, "Inferring parameters and structure of latent variable models by variational Bayes," in Proc. UAI 15, 1999.

[5] S. Watanabe, Y. Minami, A. Nakamura and N. Ueda "Variational Bayesian estimation and clustering for speech recognition," IEEE Trans. on Speech and Audio Processing, vol.12, pp.365–381, 2004.

[6] S. Watanabe, A. Sako and A. Nakamura, "Automatic determination of acoustic model topology using variational Bayesian estimation and clustering," in Proc. ICASSP 2004, vol.1, pp.813–816, 2004.

[7] J. J. Odell, "The use of context in large vocabulary speech recognition," PhD dissertation, Cambridge University, 1995.

[8] S. Young, J. J. Odell and P. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in Proc. ARPA Workshop on Human Language Technology, pp.307–312, 1994.

[9] K. Hashimoto, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda, "Bayesian context clustering using cross valid prior distribution for HMM-based speech recognition," in Proc. Interspeech, pp.936–939, 2008.

[10] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in Proc. ICASSP, pp.936–939, 2000.

[11] A. Kurematsu, K. Takeda, Y. Sagisaka, S. Katagiri, H. Kuwabara, and K. Shikano, "ATR Japanese speech database as a tool of speech recognition and synthesis," Speech Commun., vol.9, pp.357–363, 1990.

[12] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling," in Proc. ICASSP, pp.229–232, 1999.

[13] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Hidden semi-Markov model based speech synthesis," in Proc. ICSLP, pp.1185–1180, 2004.

[14] K. Shinoda and T. Watanabe, "Acoustic Modeling Based on the MDL Criterion for speech recognition," in Proc. Eurospeech, pp.99–102, 1997.