論文 / 著書情報
Article / Book Information

題目(和文)	
Title(English)	Statistical Parametric Speech Synthesis Using Local Variance and Quantized F0 Context
著者(和文)	CHUNWIJITRAVATAYA
Author(English)	Vataya Chunwijitra
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第9322号, 授与年月日:2013年9月25日, 学位の種別:課程博士, 審査員:小林 隆夫,羽鳥 好律,小池 康晴,杉野 暢彦,篠崎 隆宏
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第9322号, Conferred date:2013/9/25, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Statistical Parametric Speech Synthesis Using Local Variance and Quantized F0 Context

Vataya Chunwijitra

September 2013

Summary

This thesis describes novel tone-modeling and parameter generation approaches to generate more natural-sounding speech in statistical parametric speech synthesis. In this work, we concentrate on a major technique of statistical parametric speech synthesis approach, hidden Markov model (HMM)-based speech synthesis. In this technique, there are some issues which should be investigated and improved. Therefore, in this thesis, the work is divided into two phases including the first phase of the tone correctness improvement in tonal language and the second phase of the spectral reproducibility improvement.

In the first phase, we describe a novel approach to improving the tone correctness in speech synthesis of a tonal language based on an average-voice model trained with a corpus from nonprofessional speakers' speech. The intelligibility and naturalness of synthetic speech are degraded in HMM-based speech synthesis when using a small amount of speech data from nonprofessional speakers. There are tone disagreements between the tonal labels and the recorded speech samples when the labels are automatically generated from transcriptions. The problem with inconsistent labeling in tonal languages is crucial because incorrect tone labels affect the tone correctness of synthesized speech. Moreover, it is not an easy task to manually modify incorrect tone labels inexpensively. Therefore, we focused on reducing tone disagreements in speech data acquired from nonprofessional speakers without manually modifying the labels. To reduce the distortion in tone caused by inconsistent tonal labeling, quantized F0 symbols were utilized as the tone context to obtain an appropriate F0 model. With this technique, the tonal context label can be directly extracted from the original speech and this prevents inconsistency between speech data and F0 labels generated from

transcriptions, which affect naturalness and the tone correctness in synthetic speech. We examined two types of labeling for the tonal context using phone-based and sub-phone-based quantized F0 symbols. Subjective and objective evaluations of the synthetic voice were carried out in terms of the intelligibility of tone and its naturalness. The experimental results from both the objective and subjective tests revealed that the proposed technique could improve not only naturalness but also the tone correctness of synthetic speech under conditions where a small amount of speech data from nonprofessional target speakers was used.

In the second phase, we describe a novel approach to improving the spectral reproducibility by reducing the over-smoothing problem. In the conventional parameter generation algorithm, the resultant spectral trajectory is often excessively smoothed by the parameter tying in the model training. This causes the degradation of perceptual quality and makes the synthetic speech sounding buzzy and muffled. To alleviate the over-smoothing effect, we propose a parameter generation algorithm using a local variance (LV) model in HMM-based speech synthesis. In the proposed technique, we define LV as a feature that represents the local variation of a spectral parameter sequence and model LVs using HMMs. Context-dependent HMMs are used to capture the dependence of LV trajectories on phonetic and prosodic contexts. In addition, the dynamic features of LVs are taken into account as well as the static one to appropriately model the dynamic characteristics of LV trajectories. By introducing the LV model into the speech parameter generation process, the proposed technique can impose a more precise variance constraint for each frame than the conventional technique with a global variance (GV) model. Consequently, the proposed technique alleviates the excessive spectral peak enhancement that often occurs in GV-based parameter generation. Objective evaluation results showed that the proposed technique can generate better spectral parameter trajectories than the GV-based technique in terms of spectral and LV distortion. Moreover, the results of subjective evaluation demonstrated that the proposed technique can generate synthetic speech significantly closer to the original one than the conventional technique while maintaining speech naturalness.

Acknowledgments

I would like to thank to many people for their support, encouragement and guidance during my years as a graduate student at Tokyo Institute of Technology.

First and foremost, I wish to express my deepest gratitude to Professor Takao Kobayashi of Tokyo Institute of Technology for giving me the opportunity to study in this excellent institute. He has helped me shape my research from the beginning, guided and pushed me to get through all the inevitable research setbacks. I hereby would like to thank to all of the committee and reviewer professors for theirs kind suggestions for valuable discussions and comments regarding to my research.

I thank my colleagues at the Kobayashi laboratory for fruitful discussions about my dissertation study, and for creating such a nice work atmosphere. I would like to emphasize on the precious assistance from Dr. Takashi Nose, assistant professor of the Kobayashi Laboratory at Tokyo Institute of Technology. His substantial help in my work is deeply appreciated. Moreover, I would like to thank to friends, Thai students in Tokyo Institute of Technology for their devoting so much time and energy to evaluate our implemented speech synthesis system.

I would like to express my sincere gratitude to National Electronics and Computer Technology Center for providing us the TSynC-1 and LOTUS speech database. These speech materials are very useful to study. Over the years, I was financially supported by the Thai government science and technology scholarship. Without this support, I could not possibly complete this study.

Most of all, I would like to thank my family who have always been and continue to be there for me all the time.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 General background

Speech is the ordinary way between human beings in the communication. Moreover, speech can easily convey not only the linguistic information but also the non-linguistic one such as emotions, attitude, and speaker individuality. Therefore, it is said that speech is obviously one of the most natural, convenient, and important ways of human communication. Recently, advanced computer technologies have come into common use and play the major role in the daily life. Computers have influenced practically every field of activity such as the traffic control and information in large cities, automation in banks and railway stations, ticketing and reservation, and so on. As computers become more functional and prevalent, demands for computer technologies in speech processing area, such as speech recognition, speech understanding, speech verification, dialogue processing, and speech synthesis, are increasing for establishing more advanced human-computer interaction (HCI) using voice. There have been a great number of efforts to incorporate speech into HCI environments. Text-to-speech (TTS) synthesis is one of the efforts which have been widely developed in these decades.

TTS is a one of the key technologies in speech processing for converting an arbitrarity given text into the spoken language voice in order to transmit information from a machine to a person by voice. TTS technology is widely used in many practical applications, e.g. talking web browser, car naviga-

tion, announcements in railway stations and hospitals, response services in telecommunications, and e-mail reading. TTS technology will also be applicable to human-to-human communication with spoken language translation systems, eyes-free and/or hands-free communication or control for handicapped persons, and so on. To bring HCI closer to human-human interaction, TTS system that could generate natural sounding speech and the intelligible speech with various voice characteristics are required. In other words, the speech with various speaking voice is essential in real-life speech communication. In the past decades, TTS systems based on speech unit selection and waveform concatenation techniques, such as TD-PSOLA [5], or CHATR [6], have been proposed and shown to be able to generate natural sounding speech. The unit selection technique is becoming widely and successfully used with the increasing availability of large speech databases. However, it is not straightforward to make these systems have the ability of synthesizing speech with multiple speaker voice characteristics. One of reasons comes from the fact that the corpus-based concatenative speech synthesis always needs several large-scale speech corpora to synthesize several target speakers' voices, and consequently needs high cost and takes a lot of time to record and prepare the large scale speech data for each desired speaker. It is obviously seen that the realistic and desirable size of the speech data required for a new speaker should be as small as we can easily obtain and prepare.

Regarding the flexibility in generating speech variations, one promising approach to overcoming this problem is statistical parametric speech synthesis. This technique can produce promising results and has grown in popularity over the last few years [7]–[9]. At synthesis time, the actual segment of speech are selected from a database in the unit selection approach. On the other hands, with statistical parametric speech synthesis, statistics are obtained from a database in the training stage and fed into generative statistical models at synthesis time. According to these statistics, new speech parameter can be generated. The term parametric means that a parametric representation of the speech waveform is used, not the waveform itself.

Parametric speech synthesis based on hidden Markov models (HMMs) [7], [10] is an effective framework because of its stability and flexibility. In this

technique the speech parameters of each speech unit, such as the spectrum, fundamental frequency (F0), and phone duration, are statistically modeled and generated by using HMMs. It has already been shown that the HMM-based speech synthesis has various advantages over the concatenative speech synthesis approach. For instance, it can generate flexible characteristics of synthetic voice which can be easily modified, it can also be applied to various languages with a little modification, and it has a relatively small footprint. However, there are some issues which should be investigated and improved in HMM-based speech synthesis. The fist issue is that this approach still lacks the naturalness of the generated speech compared to concatenation speech synthesis since the synthetic speech sometimes sounding muffled. Important details of the speech waveform get lost due to too much smoothness of generated spectra, called over-smoothing problem. The second issue is tone correctness especially in tonal language. When using the low-conditioned training data in basic system, the generated tone contour is not suitable, which degrades the synthetic speech quality in terms of the tone correctness.

1.2 Scope of thesis

In this thesis, our main objective is to address two issues in the parametric speech synthesis based on hidden Markov models. We describe novel tone-modeling and parameter generation approaches to generating more natural-sounding speech.

First, we propose a tone-modeling technique using a quantized F0 context for tonal language speech synthesis based on an average voice model trained with nonprofessional speech corpus to improve the tone correctness. We focus on reducing tone disagreements in speech data acquired from nonprofessional speakers without manually modifying the labels. Since tone distortion can deteriorate not only the speech intelligibility but also the speech naturalness, the lexical tone is a suprasegmental feature formed by the basic prosodic feature, i.e., F0 [11], [12]. Therefore the tone correctness must be carefully taken into account in tonal languages. In our proposed technique, we utilize quantized F0 context as the tonal context in order to obtain an appropriate F0 model. With this technique, the tonal context can be extracted from real

speech directly and this leads to preventing the inconsistency between speech data and tone labels generated from transcription, which affects the naturalness and tone correctness in synthetic speech. In this thesis, we propose two methods of tone context labeling using the quantized F0 symbols based on phone and sub-phone boundaries. Subjective and objective evaluations of the synthetic voice are carried out in terms of the intelligibility of tone and its naturalness. The experimental results from both objective and subjective tests will reveal that the proposed technique can improve not only naturalness but also the tone correctness of synthetic speech under conditions where a small amount of speech data from nonprofessional target speakers is used.

Second, we describe a technique for modeling local variance (LV) of speech features and propose a novel parameter generation algorithm using the LV model for the HMM-based speech synthesis. In the proposed technique, we define LV as a feature that represents the local variation around each frame of the spectral features and model them using HMMs. Context-dependent HMMs are used to capture the dependence of LV trajectories on phonetic and prosodic contexts. In addition, to appropriately model the dynamic characteristics of LVs, we take into account the dynamic features of LVs as well as the static one. In the parameter generation process, a spectral parameter sequence is estimated so as to maximize a target function, in which conventional HMMs and LV models are combined. By using the LV models, the proposed technique can impose a more precise variance restriction in the parameter generation than the conventional technique where the global variance (GV) [13], [14] model is used. Consequently, the proposed technique alleviates the excessive spectral peak enhancement that often occurs in GV-based parameter generation. Subjective experimental results also show that the proposed technique significantly improve the reproducibility of the synthetic speech than the conventional one while maintaining speech naturalness.

This thesis is organized as follows. At first, the principle of the HMM-based statistical parametric speech synthesis approach is described in Chapter 2. We also explain two drawbacks of this approach including the tone correctness in tonal language and the spectral reproducibility. As for the first issue of study, Chapter 3 presents the tone-modeling technique using a quantized F0 context to alleviate the tone disagreement problem in tonal lan-

guage. Chapter 4 addresses the second issue on the over-smoothing problem in synthetic speech. The LV model is incorporated into the parameter generation algorithm to improve the spectral reproducibility. Finally, Chapter 5 gives conclusions of this thesis and future work.

Chapter 2

Statistical Parametric Speech Synthesis

This chapter describes a text-to-speech (TTS) framework with hidden Markov model (HMM) based statistical parametric speech synthesis. In HMM-based speech synthesis approach, the parameters of a speech unit, such as spectrum, fundamental frequency (F0), and phoneme duration are statistically modeled and generated by using HMMs, When synthesizing speech, speech spectral parameter sequences are generated from HMMs directly based on maximum likelihood (ML) criterion. In this chapter, we briefly explain the basic structure and algorithms of the HMM-based TTS system.

2.1 Introduction

Text-to-speech is a technique to generate a spoken sound from an arbitrarily given normal text automatically. This technology allows interaction of the application with the user on a personal level. It allows users to receive information without having to take their eyes off whatever they are doing. Up until now, there are several approaches to constructing the TTS systems such as unit selection [15], diphone synthesis [16], and HMM-based synthesis [7]. In these approaches, the unit selection is most successful one in recent business applications. The advantage of the unit selection approach is its quality.

This technique simply stores the speech corpus itself; this implies that the entire corpus or just selected parts of it. The synthetic speech is generated by concatenating the speech waveform units selected from speech corpus so as to minimize the target and concatenative costs. As a result, by using a huge amount of target speaker's speech data, high quality synthetic speech can be obtained. However, the cost of database construction tends to very expensive and this fact makes it difficult to synthesize various speakers' voices.

In contrast to the selection of actual instances of speech from a database, a model-based speech synthesis approach does not store any speech but it learns a model, generally called a statistical parametric model, from the speech database during the training phase. The model is parametric because it describes the speech using parameters, rather than stored exemplars. It is statistical because it describes those parameters using statistics, e.g., means and variances of probability density functions (pdf), which capture the distribution of parameter values found in the training data.

Historically, the starting point for statistical parametric speech synthesis was the success of the HMM for automatic speech recognition. The availability of effective and efficient learning algorithms (e.g., expectation-maximization algorithm), automatic methods for model complexity control (e.g., parameter tying) and computationally efficient search algorithms (e.g., Viterbi search) make the HMM a powerful model. The performance of the model, which in speech recognition is measured using word error rates and in speech synthesis by listening tests, depends critically on choosing an appropriate configuration. The two most important aspects of this configuration are the parameterization of the speech signal (the "observations" of the model, in HMM terminology) and the choice of modeling unit.

In recent years, parametric speech synthesis based on HMMs have been increasing thanks to its stability and flexibility. Specifically, this approach allows us not only to produce smooth and stable speech under a small footprint but also to add more variations to synthetic speech by using a variety of techniques. By modeling sequences of speech features using HMMs, we can generate synthetic speech statistically from HMMs. Besides, once HMMs are trained, the various operations, e.g, adaptation, interpolation, control techniques for speaker characteristics, emotional expressions and speaking styles

can be applied. In this chapter, we give a brief overview of the HMM-based statistical parametric speech synthesis [1] and its advantages and drawbacks, which is important to understand the techniques proposed in this thesis.

2.2 Statistical parametric speech synthesis based on HMM: System overview

In a typical statistical parametric speech synthesis system, we first extract parametric representations of speech including spectral and excitation parameters from a speech database and then model them by using a set of generative models (e.g., HMMs). A maximum likelihood criterion is usually used to estimate the model parameters. We then generate speech parameters for a given text to be synthesized from the set of estimated models. Finally, a speech waveform is reconstructed from the parametric representations of speech. Although any generative model can be used, HMMs have been widely used. Statistical parametric speech synthesis with HMMs is commonly known as HMM-based speech synthesis [7].

In the HMM-based speech synthesis, each speech unit, generally phoneme unit, is modeled using an HMM. To model the speech features, i.e., spectrum, F0, and phone duration, appropriately, the standard HMM has been extended to the HMM with multi-space probability distribution (MSD) [17] and explicit duration modeling. MSD is used to model the F0 sequences which have scalar values in voiced regions and undefined values in unvoiced regions. The HMM with explicit duration modeling is called the hidden semi-Markov model (HSMM) [18]. Sequence of speech features is generated from HSMMs based on an ML criterion.

Figure 2.1 shows a basic structure of an HMM-based statistical parametric speech synthesis system. The system mainly consists of training and synthesis stages.

In the training stage, context dependent phoneme HSMMs are trained using a speech database. Spectrum and excitation parameter (F0) are extracted at each analysis frame as the static features from the speech database and modeled by multi-stream HMMs in which output pdfs for the spectral

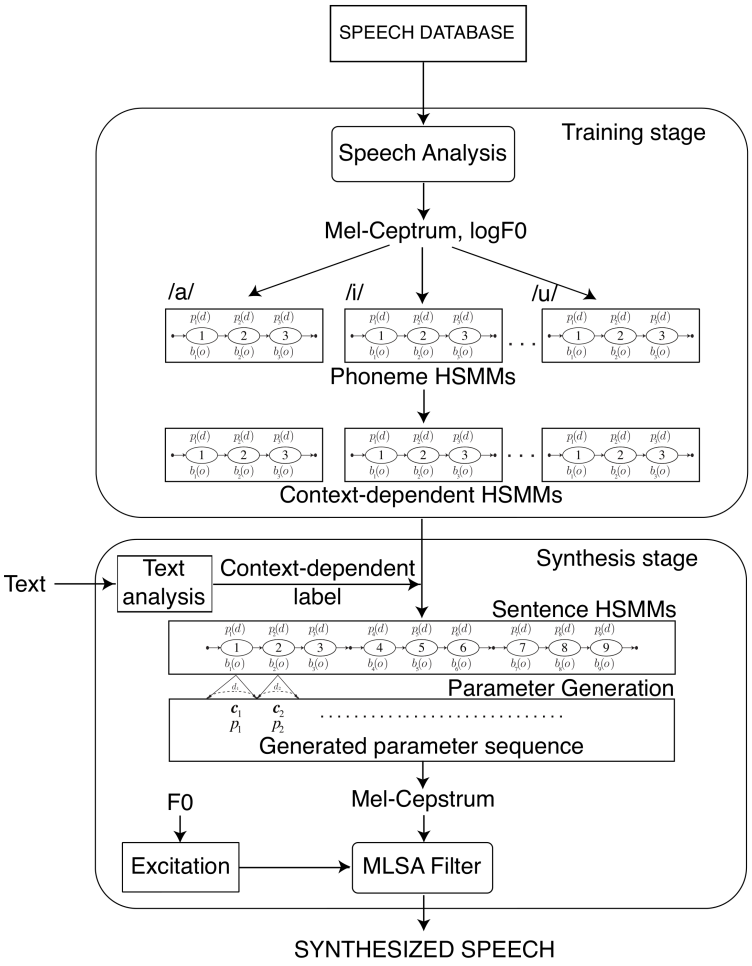


Figure 2.1: HMM-based speech synthesis system.

and F0 parts are modeled using a continuous probability distribution and the multi-space probability distribution [17], [19], respectively. Moreover, to directly model and control phone durations, we utilize a framework of hidden semi-Markov model [18], [20], [21] which is an HMM having explicit state duration distributions instead of the transition probabilities. To model variations in the spectrum, F0, and duration, we take into account phonetic, prosodic, and linguistic contexts, such as phoneme identity contexts, stress-related contexts, and locational contexts. Then, the decision-tree-based context clustering technique [22] is applied separately to the spectral, F0, and

duration parts of the context-dependent phoneme HMMs. In the clustering technique, a decision tree is automatically constructed based on the minimum description length (MDL) criterion [23]. Thereafter, we perform re-estimation processes of the clustered context-dependent phoneme HMMs using the Baum-Welch (EM) algorithm. Finally, state durations are modeled by a multivariate Gaussian distribution [24], and the state clustering technique is also applied to the state duration models.

In the synthesis stage, an arbitrarily given text is first transformed into a sequence of context-dependent phoneme labels. Based on the label sequence, a sentence HMM is constructed by concatenating context-dependent phoneme HMMs. From the sentence HMM, spectral and F0 parameter sequences are obtained based on the ML criterion [25] in which phoneme durations are determined using state duration pdfs. Finally, the output speech waveform is synthesized from the generated mel-cepstral and F0 parameter sequences by using a Mel Log Spectral Approximation (MLSA) filter [26], [27].

2.3 Multi-space probability distribution HMM

To generate the synthetic speech, it is necessary to model and generate F0 patterns as well as spectral sequences. However, we cannot model the F0 patterns by conventional discrete or continuous HMMs, because the F0 values are not defined in unvoiced regions, i.e., the observation sequence of an F0 pattern is composed of one-dimensional continuous values in voiced region and a discrete symbol which represents unvoiced region. Therefore, the multi-space probability distribution HMM (MSD-HMM) [17], [19] was proposed for F0 pattern modeling and generation.

In the MSD-HMM, the observation sequence of F0 pattern is viewed as a mixed sequence of outputs from a one-dimensional space Ω_1 and a zero-dimensional space Ω_2 which correspond to voiced and unvoiced regions, respectively. Each space has the space weight w_g ($\sum_{g=1}^{2} w_g = 1$). The space Ω_1 has a one-dimensional normal pdf $\mathcal{N}_1(\boldsymbol{x})$. Meanwhile, the space Ω_2 has only one sample point. Thus, an F0 observation $\boldsymbol{o}$ consists of a continuous

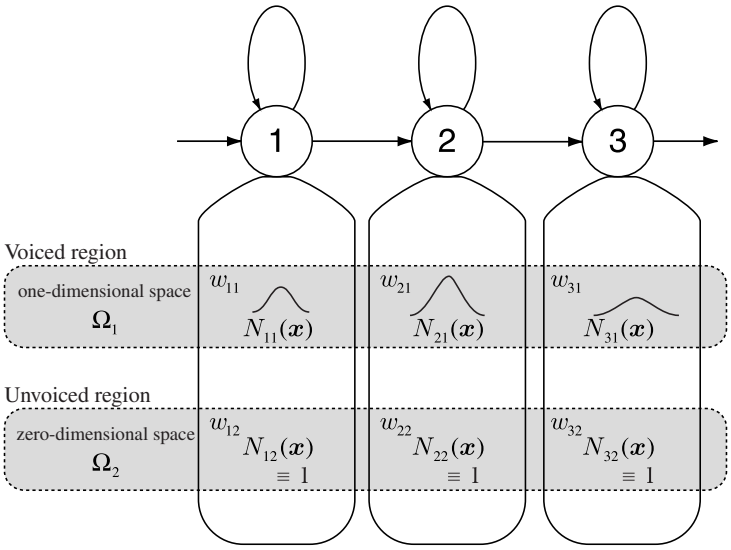


Figure 2.2: MSD-HMM [1].

random variable $\boldsymbol{x}$ and a set of space indices X, that is,

$$\boldsymbol{o} = (X, \boldsymbol{x}) \tag{2.1}$$

where $X = \{1\}$ for voiced region and $X = \{2\}$ for unvoiced region. Subsequently, the observation probability of $\boldsymbol{o}$ is defined by

$$b(\boldsymbol{o}) = \sum_{g \in S(\boldsymbol{o})} w_g \mathcal{N}_g(V(\boldsymbol{o})) \tag{2.2}$$

where $V(\boldsymbol{o}) = \boldsymbol{x}$ and $S(\boldsymbol{o}) = X$. Note that, although $\mathcal{N}_2(\boldsymbol{x})$ does not exist for Ω_2, $\mathcal{N}_2(\boldsymbol{x}) \equiv 1$ is defined for simplicity of notation. An example of the MSD-HMM for F0 pattern modeling is shown in Figure 2.2.

Using the MSD-HMM, we can model voiced and unvoiced observations of F0 in a unified model without any heuristic assumption [17], [19]. In addition, spectrum and F0 can be modeled simultaneously by multi-stream MSD-HMM, where spectral part is modeled by continuous probability distribution, and F0 part is modeled by MSD as shown in Figure 2.3. In this figure, $\boldsymbol{c}_t$, X_t^p, and $\boldsymbol{x}_t^p$ represent the spectral parameter vector, a set of space indices of F0, and F0 parameter at time t, respectively, and Δ and Δ^2 represent the delta and delta-delta parameters, respectively.

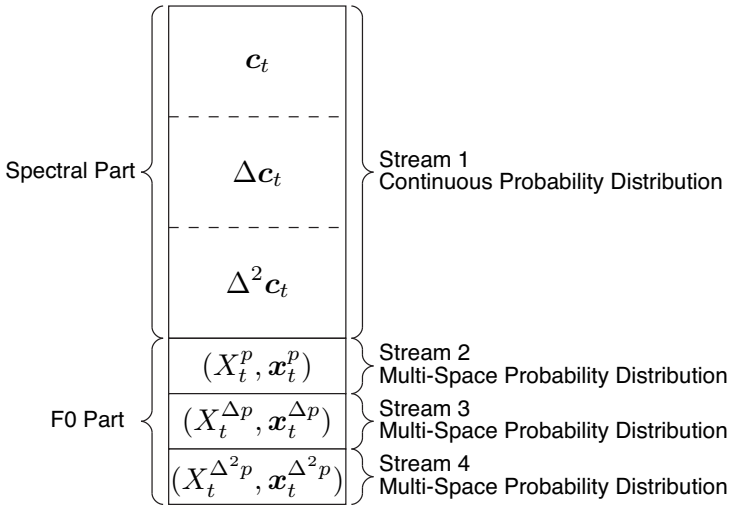


Figure 2.3: Observation vector [1].

2.4 Hidden semi-Markov model

In the original HMM, the phoneme duration is represented by the transition probabilities. This is not suitable for the speech synthesis because the explicit duration modeling is required when generating speech parameter sequences from a given text [28]. To solve this problem, we employ a framework of HSMM [20], where the model has explicit state duration distributions instead of the transition probabilities to directly model and control phone durations as shown in Figures 2.4 and 2.5.

An N-state left-to-right HSMM λ without skip paths is specified by state output pdf $\{b_i(\cdot)\}_{i=1}^N$ and state duration pdf $\{p_i(\cdot)\}_{i=1}^N$. In this study, we assume that the i-th state output and duration pdfs are Gaussian distributions characterized by mean vector $\boldsymbol{\mu}_i$ and diagonal covariance matrix $\boldsymbol{\Sigma}_i$, and mean m_i and variance σ_i^2, respectively,

$$b_i(\boldsymbol{o}) = \mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \tag{2.3}$$

$$p_i(d) = \mathcal{N}(d; m_i, \sigma_i^2) \tag{2.4}$$

where $\boldsymbol{o}$ is L-dimensional observation vector and d is duration staying in the state i.

The training of the parameter set λ based on maximum likelihood crite-

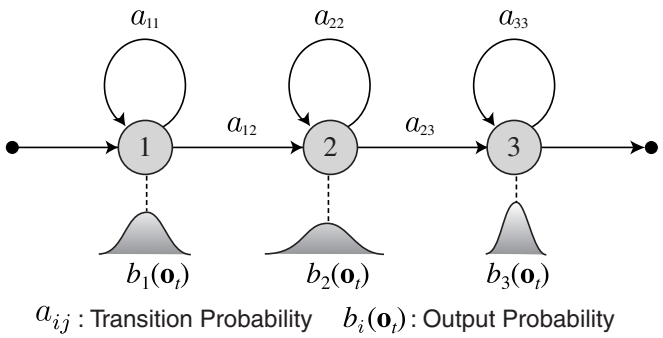


Figure 2.4: Hidden Markov model [1].

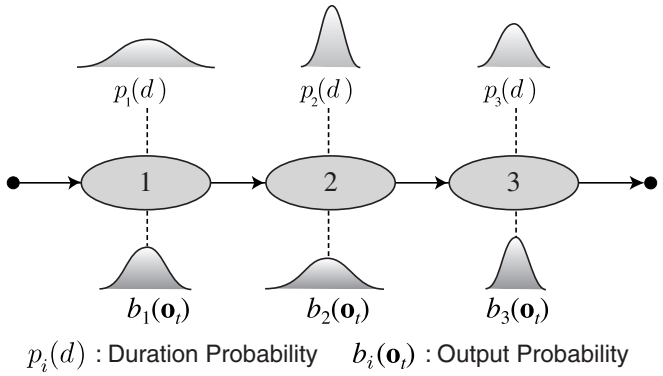


Figure 2.5: Hidden semi-Markov model [1].

rion can be formulated as follows:

$$\hat{\lambda} = \arg\max_{\lambda} P(\boldsymbol{O}|\lambda). \tag{2.5}$$

Re-estimation formulas based on the Baum-Welch algorithm of the parameter set λ are given by [18]

$$\overline{\boldsymbol{\mu}}_i = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)\sum_{s=t-d+1}^{t}\boldsymbol{o}_s}{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)\,d} \tag{2.6}$$

$$\overline{\boldsymbol{\Sigma}}_i = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)\sum_{s=t-d+1}^{t}(\boldsymbol{o}_s-\overline{\boldsymbol{\mu}}_i)(\boldsymbol{o}_s-\overline{\boldsymbol{\mu}}_i)^\top}{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)\,d} \tag{2.7}$$

$$\overline{m}_i = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)\,d}{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)} \tag{2.8}$$

$$\overline{\sigma}_i^2 = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)\,(d-\overline{m}_i)^2}{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(i)} \tag{2.9}$$

where $\cdot^\top$ denotes matrix transpose, and $\gamma_t^d(i)$ is the probability generating observation sequence $\boldsymbol{o}_{t-d+1}, \cdots, \boldsymbol{o}_t$ at the i-th state and defined by

$$\gamma_t^d(i) = \frac{1}{P(\boldsymbol{O}|\lambda)} \sum_{\substack{j=1 \\ j \neq i}}^{N} \alpha_{t-d}(j)\, p_i(d) \prod_{s=t-d+1}^{t} b_i(\boldsymbol{o}_s) \beta_t(i) \tag{2.10}$$

where $\alpha_t(i)$ and $\beta_t(i)$ are the forward and backward probabilities in the state i at time t.

2.5 Decision-tree-based context clustering

In the HMM-based speech synthesis system, parameter sequences of particular speech unit (e.g., phoneme) vary depending on phonetic context. To reduce the variations, in the HMM-based speech synthesis system, we utilize speech units considering prosodic and linguistic contexts, such as syllable, phrase, part of speech, and sentence information, to model suprasegmental features in prosodic feature appropriately. Ideally, all the different realizations of speech units should be available in a database. However, it is impossible to prepare training data that cover all possible context dependent units. Moreover, there is great variation in the frequency of appearance of each context dependent unit. To alleviate these problems, we employ a decision-tree-based context clustering algorithm [22] to cluster HMM states and share model parameters among states in each cluster.

An example of the decision tree is shown in Figure 2.6. Each non-terminating node has a context related question, such as R-silence? ("is the succeeding phoneme a silence?") or L-voiced? ("is the preceding phoneme a voiced phoneme?"), and two child nodes representing "yes" and "no" answers to the question. Several questions are concatenated until the final leaf node is reached. All final leaf nodes have state output distributions. Using the decision-tree-based context clustering, model parameters of the speech units for the unseen contexts can be obtained, because any context can reach one of the leaf nodes by going down the tree starting from the root node then selecting the next node depending on the answer about the current context.

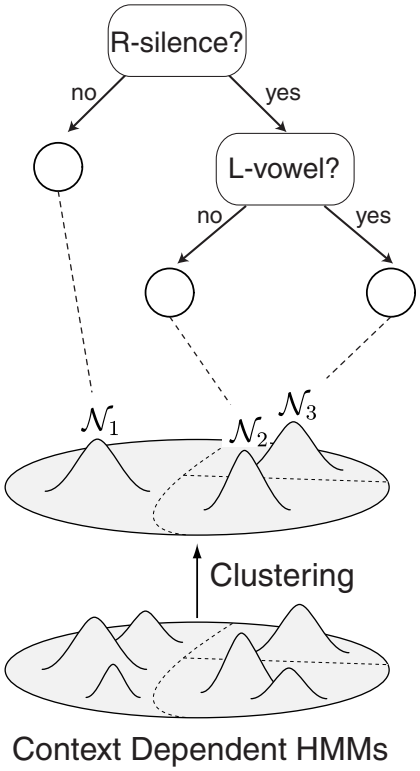


Figure 2.6: An example of decision tree [1].

2.6 Speech parameter generation from HMM

To model and synthesize speech parameter, we use dynamic features as well as static features. Let us assume a D-dimensional speech parameter vector $\boldsymbol{c}_t = [c_t(1), ..., c_t(d), ..., c_t(D)]^\top$ at the t-th frame. A feature vector $\boldsymbol{o}_t = [\boldsymbol{c}_t^\top, \Delta^{(1)}\boldsymbol{c}_t^\top, \Delta^{(2)}\boldsymbol{c}_t^\top]^\top$ consists of static and dynamic feature vectors and is used as an observation vector to maintain the property of smoothness for the generated parameter sequence. The dynamic feature vectors, $\Delta^{(1)}\boldsymbol{c}_t$ and $\Delta^{(2)}\boldsymbol{c}_t$, are calculated frame by frame as follows:

$$\Delta^{(n)}\boldsymbol{c}_t = \sum_{\tau=-L_-^{(n)}}^{L_+^{(n)}} \boldsymbol{w}^{(n)}(\tau)\boldsymbol{c}_{t+\tau}, \qquad n = 1, 2. \tag{2.11}$$

The sequences of $\boldsymbol{o}_t$ and $\boldsymbol{c}_t$ are written in a vector form as $\boldsymbol{o} = [\boldsymbol{o}_1^\top, ..., \boldsymbol{o}_t^\top, ..., \boldsymbol{o}_T^\top]^\top$ and $\boldsymbol{c} = [\boldsymbol{c}_1^\top, ..., \boldsymbol{c}_t^\top, ..., \boldsymbol{c}_T^\top]^\top$, respectively. The relation-

ship between these two vectors is represented by linear transformation as follows:

$$\boldsymbol{o} = \boldsymbol{W}\boldsymbol{c} \tag{2.12}$$

where $\boldsymbol{W}$ is a $3DT \times DT$ matrix [25] written as

$$\boldsymbol{W} = [\boldsymbol{W}_1, ..., \boldsymbol{W}_t, ..., \boldsymbol{W}_T]^\top \otimes \boldsymbol{I}_{D\times D} \tag{2.13}$$

$$\boldsymbol{W}_t = [\boldsymbol{w}_t^{(0)}, \boldsymbol{w}_t^{(1)}, \boldsymbol{w}_t^{(2)}] \tag{2.14}$$

$$\begin{aligned}\boldsymbol{w}_t^{(n)} = [\overbrace{0, ..., 0}^{1\text{st}}, \overbrace{w^{(n)}(-L_-^{(n)})}^{(t-L_-^{(n)})\text{-th}}, ..., \overbrace{w^{(n)}(0)}^{t\text{-th}}, ...,\\ \overbrace{w^{(n)}(L_+^{(n)})}^{(t+L_+^{(n)})\text{-th}}, 0, ..., \overbrace{0}^{T\text{-th}}]^\top, \quad n = 0, 1, 2.\end{aligned} \tag{2.15}$$

Here, $L_-^{(0)} = L_+^{(0)} = 0$ and $w^{(0)}(0) = 1$. The operator $\otimes$ denotes the Kronecker product. The matrix $\boldsymbol{I}_{D\times D}$ denotes the $D \times D$ identity matrix.

The observation sequence vector $\boldsymbol{o}$ is modeled by context-dependent phone HMMs whose parameter set is denoted by $\boldsymbol{\lambda}$. The pdf for the sequence $\boldsymbol{o}$, given an HMM state sequence $\boldsymbol{q} = \{q_1, ..., q_t, ..., q_T\}$, is typically represented by a single multivariate Gaussian pdf as follows:

$$\begin{aligned}P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}) &= \mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}_q, \boldsymbol{U}_q)\\ &= \frac{1}{\sqrt{(2\pi)^{3DT}|\boldsymbol{U}_q|}} \exp\left\{-\frac{1}{2}(\boldsymbol{o} - \boldsymbol{\mu}_q)^\top \boldsymbol{U}_q^{-1}(\boldsymbol{o} - \boldsymbol{\mu}_q)\right\}\end{aligned} \tag{2.16}$$

where $\boldsymbol{\mu}_q$ and $\boldsymbol{U}_q$ are a $3DT \times 1$ mean vector and a $3DT \times 3DT$ covariance matrix, respectively which are given by

$$\boldsymbol{\mu}_q = [\mu_{q_1}^\top, ..., \mu_{q_t}^\top, ..., \mu_{q_T}^\top]^\top \tag{2.17}$$

$$\boldsymbol{U}_q = \mathrm{diag}[U_{q_1}, ..., U_{q_t}, ..., U_{q_T}]. \tag{2.18}$$

In the synthesis stage, a time sequence of static feature vectors $\hat{\boldsymbol{c}}$ is determined by maximizing the HMM likelihood under the condition as follows:

$$\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{c}} P(\boldsymbol{o}, \boldsymbol{\lambda}) \qquad \text{subject to } \boldsymbol{o} = \boldsymbol{W}\boldsymbol{c} \tag{2.19}$$

and then the HMM state sequence and the static feature vector sequence are sequentially determined. A suboptimal HMM state sequence is determined

by

$$\hat{\boldsymbol{q}} = \arg\max_{\boldsymbol{c}} P(\boldsymbol{q}, \boldsymbol{\lambda}). \tag{2.20}$$

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

$$\hat{\boldsymbol{c}} = \arg\max P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}) \qquad \text{subject to } \boldsymbol{o} = \boldsymbol{W}\boldsymbol{c}. \tag{2.21}$$

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

$$\begin{aligned} \mathcal{L}_q &= \log P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda}) && (2.22) \\ &\propto -\frac{1}{2}\boldsymbol{o}^\top \boldsymbol{U}_q^{-1}\boldsymbol{o} + \boldsymbol{o}^\top \boldsymbol{U}_q^{-1}\boldsymbol{\mu}_q \\ &= -\frac{1}{2}\boldsymbol{c}^\top \boldsymbol{W}^\top \boldsymbol{U}_q^{-1}\boldsymbol{W}\boldsymbol{c} + \boldsymbol{c}^\top \boldsymbol{W}^\top \boldsymbol{U}_q^{-1}\boldsymbol{\mu}_q \\ &= -\frac{1}{2}\boldsymbol{c}^\top \boldsymbol{R}_q\boldsymbol{c} + \boldsymbol{c}^\top \boldsymbol{r}_q && (2.23) \end{aligned}$$

where

$$\boldsymbol{R}_q = \boldsymbol{W}^\top \boldsymbol{U}_q^{-1}\boldsymbol{W} \tag{2.24}$$

$$\boldsymbol{r}_q = \boldsymbol{W}^\top \boldsymbol{U}_q^{-1}\boldsymbol{\mu}_q. \tag{2.25}$$

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

$$\hat{\boldsymbol{c}_q} = \boldsymbol{P}_q\boldsymbol{r}_q \tag{2.26}$$

$$\boldsymbol{P}_q = \boldsymbol{R}_q^{-1}. \tag{2.27}$$

Since the matrix $\boldsymbol{P}_q$ is generally full owing to the inverse of the band matrix $\boldsymbol{R}_q$, the state output pdf 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 pdf sequence corresponding to discrete state sequences so that the generated trajectories exhibit suitable static and dynamic properties.

2.7 Model adaptation and average voice model

In this section, the HSMM-based MLLR adaptation and the average voice model is briefly described [1]. In general, it is desirable that speech synthesis systems have the ability to synthesize speech with arbitrary multiple speakers' voice characteristics. For instance, considering the speech translation systems which are used by a number of speakers simultaneously, it is necessary to reproduce input speakers' characteristics to make listeners to distinguish speakers of the translated speech. Another instance, in the spoken dialog systems with multiple agents, each agent should have his or her own speaker characteristics.

Since the HMM-based speech synthesis approach uses HMMs as the speech units in both modeling and synthesis, we can easily change voice characteristics of synthetic speech by transforming HMM parameters appropriately. In the training of HMMs, a sufficient amount of target speaker's data is required to synthesize natural sounding speech. To train the HMMs using only a small amount of speech data of the target speaker, several model adaptation techniques have been proposed. In the model adaptation, average voice model trained with multiple speakers' data is utilized to improve the adaptation performance. The clustering technique for the training of average voice model have also been proposed based on shared decision tree.

2.7.1 HSMM-based MLLR adaptation

In maximum likelihood linear regression (MLLR)-based adaptation, it is assumed that the mean parameters of the target speaker's model is expressed by the linear regression of those of the average voice model. In MLLR-based adaptation for HSMM [29], mean parameters of state output and duration pdfs of the target speaker are obtained by linearly transforming those of an initial model as illustrated in Fig. 2.7.

$$b_i(\boldsymbol{o}) = \mathcal{N}(\boldsymbol{o}; \boldsymbol{\zeta}\boldsymbol{\mu}_i + \boldsymbol{\epsilon}, \boldsymbol{\Sigma}_i) \tag{2.28}$$

$$= \mathcal{N}(\boldsymbol{o}; \boldsymbol{W}\boldsymbol{\xi}_i, \boldsymbol{\Sigma}_i) \tag{2.29}$$

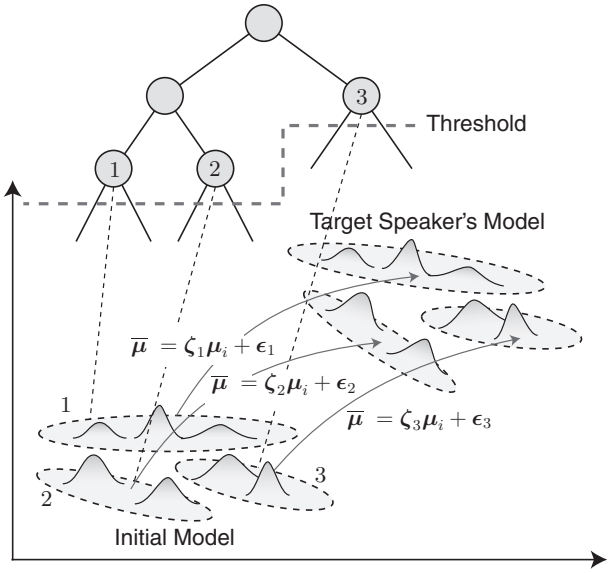


Figure 2.7: Maximum likelihood linear regression [1].

$$p_i(d) = \mathcal{N}(d; \chi m_i + \nu, \sigma_i^2) \tag{2.30}$$
$$= \mathcal{N}(d; \boldsymbol{X}\boldsymbol{\phi}_i, \sigma_i^2) \tag{2.31}$$

where $\boldsymbol{W} = [\boldsymbol{\zeta}, \boldsymbol{\epsilon}] \in \mathcal{R}^{L \times (L+1)}$ and $\boldsymbol{X} = [\chi, \nu] \in \mathcal{R}^{1 \times 2}$ are the transformation matrices which transform extended mean vectors $\boldsymbol{\xi}_i = [\boldsymbol{\mu}_i^\top, 1]^\top \in \mathcal{R}^{L+1}$ and $\boldsymbol{\phi}_i = [m_i, 1]^\top \in \mathcal{R}^2$, respectively. $\boldsymbol{\zeta}$ and $\boldsymbol{\epsilon}$ are $L \times L$ matrix and L-dimensional vector, respectively, and both χ and ν are scalar variables. The transformation matrices are obtained based on maximum likelihood estimation using an EM algorithm. In general, the adaptation data is very small, and it is difficult to estimate the transformation matrices for each distributions. Therefore the parameter tying is conducted for the reduction of the number of parameters to be estimated. Tying topology and the number of the multiple transformation matrices is determined based on the tree structure of distributions obtained in the training of initial HSMMs.

2.7.2 Average voice model

As the initial model of the model adaptation in speech synthesis, the average voice model, which is the HSMMs trained using multiple speakers' speech

data and having the average characteristics of them, is often utilized for reducing dependency of speaker characteristics of the initial model. A reason to use the average voice model is that the synthetic speech generated from speaker-adapted model is sensitive to the characteristics of the initial model and the average voice model can reduce the dependency on the speaker characteristics of initial model. In addition, the average voice model can utilize a large variety of contextual information included in the several speakers' speech database as a prior information for the speaker adaptation and provide robust basis useful for synthesizing speech of the new target speaker. As a result, synthetic speech of the target speaker can be obtained robustly even if adaptation data of the target speaker is very small.

For the training of average voice model, an effective parameter tying algorithm called shared-decision-tree-based context clustering (STC) [30] has been proposed. When using the decision-tree-based context clustering [22] for the parameter tying of the average voice model, the nodes of the decision tree do not always have training data of all speakers, and some nodes have data of only one speaker. This speaker-biased node causes degradation of quality of average voice and synthetic speech after speaker adaptation, especially in prosody. In contrast, STC constructs a decision tree so that every node always has the data of all speakers. In other words, there is no node lacking one or more training speakers' data.

When training data of each training speaker differs widely, the distributions of average voice model often have bias depending on speaker and/or gender and this will degrade the quality of synthetic speech. Therefore, to reduce the influence of speaker dependence, the speaker adaptive training (SAT) and STC are incorporated into the training procedure of average voice model [31]. Specifically, STC is used for clustering distributions of spectrum, F0, and state duration, thereafter SAT is used for re-estimation of parameters of spectrum and F0. The training data for the average voice model is consisted of speech data from several speakers. If the normal model from training method is applied directly to the average voice model directly, the model parameters of the average voice model are affected by the influence of speaker differences of the training speakers. SAT is a kind of the speaker normalization algorithm for normalizing the influence of the large difference and

the speaker dependence among the training speakers in the speech database.

2.8 Drawbacks and refinements

The biggest drawback with statistical parametric synthesis against unit-selection synthesis is the quality of synthesized speech. In this thesis, we investigate the following issues that can be identified to degrade the performance of a HMM-based synthesis system.

2.8.1 Tone correctness in tonal language

Tonal language is a language in which pitch is used as a part of speech, changing the meaning of a word. In the tonal language, tone is the term used to describe the use of pitch patterns to distinguish individual words or the grammatical forms of words, such as the singular and plural forms of nouns or different tenses of verbs. In the simplest cases, each syllable of a language with tones will have its own characteristic tonal pattern, which may be a relatively flat pitch at a particular level, or may involve the pitch rising or falling over the duration of the syllable. When the pitch has a moving pattern of this sort, the tone is described as a contour tone.

In tonal language, the tone (F0) contour is associated with some high-level prosodic features with larger scope than a phone, such as tones of a syllable, stress in a phrase, prosodic boundary in a sentence, etc. The tone contours of utterances should include these local tonal features in addition to the sentential intonation corresponding to syntactic/utterance structures. This situation makes tone movements of tonal language like Thai, Mandarin Chinese, and standard Yoruba, more complicated than non-tonal languages like English, Japanese, and so on. Therefore, control of tone contours (together with other prosodic features) becomes an important issue in tonal language speech synthesis. Although the quality of synthetic speech has been largely improve, there still remain problems if we view from the prosodic features. Or we should say that the high quality in each sound emerges problems in prosodic features.

In the conventional HMM-based speech synthesis system, the generated

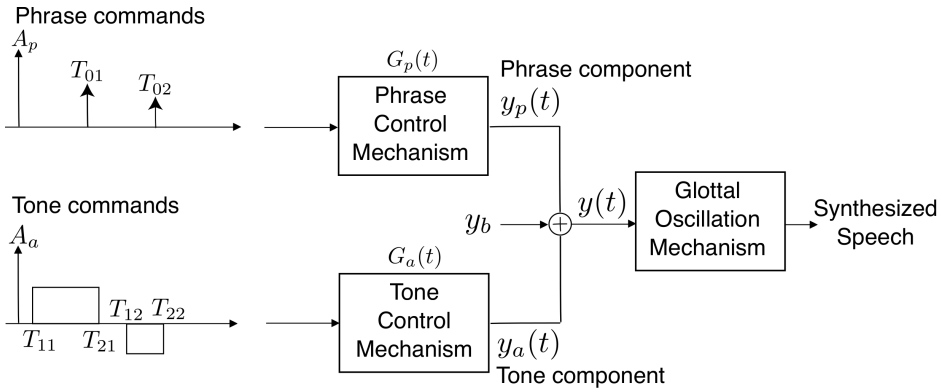


Figure 2.8: Block diagram of the Fujisaki model for synthesizing F0 contours [2].

tone contour especially in tonal language is not suitable. This degrade the synthetic speech quality in terms of the tone correctness. Tone modeling for speech synthesis aims at providing proper tone information, e.g., tone contour, to generate natural synthetic speech from input text. The tone modeling has been studied in three levels, including an acoustic level, a perceptual level, and a linguistic level [32]. In this study, we concentrate on the acoustic level which is based on capturing and modeling the acoustic signal directly. Several limitations of intonation modeling have been described in the acoustic level. The Fujisaki model [33] is an attractive approach since it explains the behavior of F0 contours by separating the global intonation contour from fine accentual or tonal structures. Another class of F0 modeling such as T-Tilt models [3], [4] is based mainly on modeling F0 in local context.

2.8.1.1 Fujisaki model

The Fujisaki model [33] is a well-known mathematical model, which describes the generating process of the whole F0 contour of a speech utterance. The remarkable feature of the Fujisaki model is that it consists of physiologically and physically meaningful parameters, called the phrase and tone commands, and is able to fit F0 contours of real speech well when they are chosen appropriately.

Block diagram of Fujisaki model for tonal language is illustrated in Figure

2.8. The phrase commands are assumed to be impulse which are applied to the phrase control mechanism and generate the phrase components. Tone commands are step function in both positive and negative polarities that are applied to tone control mechanism to generate the tone components.

In this technique, let us assume that an F0 contour on a logarithmic scale $y(t)$, where t is time, is the superposition of three components consisting a phrase component $y_p(t)$, an tone component $y_a(t)$, and a base component y_b. Mathematically, the F0 contour $y(t)$ of an utterance generated from an extension of the Fujisaki model for tonal languages is shown as follow [34]:

$$y(t) = y_b + y_p(t) + y_a(t). \tag{2.32}$$

The base component y_b is a base frequency of F0 value related to the lower bound of each speakers F0. The phrase component $y_p(t)$ consists of the pitch variations over the duration of the prosodic units, and the tone component $y_a(t)$ consists of the pitch variations in syllables.

$$y_p(t) = \sum_{i=1}^{I} A_{pi} G_p(t - T_{0i}) \tag{2.33}$$

$$y_a(t) = \sum_{j=1}^{J} A_{aj} [G_a(t - T_{1j}) - G_a(t - T_{2j})] \tag{2.34}$$

where

$$G_p(t) = \begin{cases} \alpha^2 t e^{-\alpha t} & (t \geq 0) \\ 0 & (t < 0) \end{cases} \tag{2.35}$$

$$G_a(t) = \begin{cases} [1 - (1 + \beta t) e^{-\beta t}] & (t \geq 0) \\ 0 & (t < 0). \end{cases} \tag{2.36}$$

$G_p(t)$ represents the impulse response function of phrase control mechanism. $G_a(t)$ represents the step response function of tone control mechanism. α and β are time constant parameters in phrase generation and tone generation, which are known to be almost constant within an utterance as well as across utterances for a particular speaker. T_{0i} and A_{pi} denote the i^{th} phrase command time and its amplitude, respectively. T_{1j}, T_{2j}, and A_{aj} are onset time, offset time, and amplitude of the j^{th} tone command, respectively. I

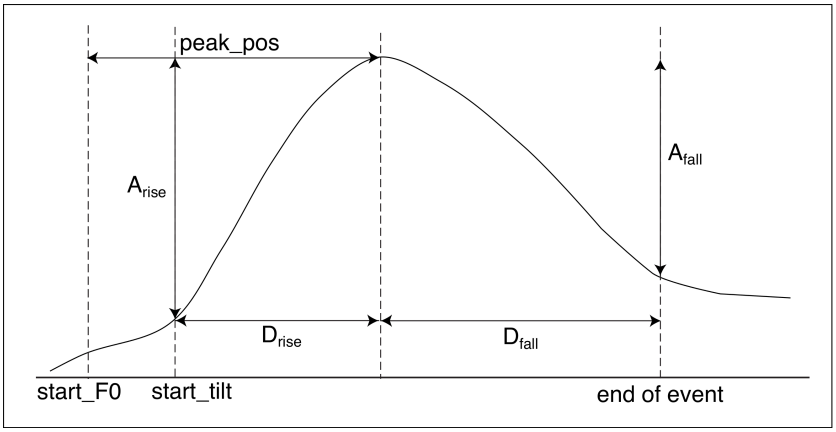


Figure 2.9: Parameterization in the TTilt model [3], [4].

and J are number of phrase and tone commands, respectively. The synthesized F0 contour $y(t)$ is passed to glottal oscillation mechanism to generate the naturalness of utterances in continuous speech synthesis system.

2.8.1.2 T-Tilt model

The T-Tilt model [3], [4] is one of parameterization approaches that was modified from the Tilt intonation model [35] to better work for tonal languages. In tonal languages, the F0 movement has often been modeled on the basis of syllable units. Figure 2.9 demonstrates the T-Tilt model parameterizes. The T-Tilt model consists of eight continuous value parameters forming the F0 contour of a syllable [3], [4] including

- start_F0: the F0 at the starting point of the syllable,
- start_tilt: the starting time of the Tilt in the syllable,
- event_amp: the summation of absolute rising (A_{rise}) and falling (A_{fall}) amplitudes (negative for the valley F0 shape),
- event_dur: the summation of rising(D_{rise}) and falling(D_{fall}) duration,
- peak_pos: the duration distance between the starting point of the syllable to the peak of the Tilt,

Table 2.1: The type of F0 shape for T-Tilt modeling [3], [4].

Label	Shape
R	rising hill
R+	rising valley
F	falling hill
F+	falling valley
RF	rising hill followed by falling hill
RF+	rising valley followed by falling valley
FR	falling hill followed by rising hill
FR+	falling valley followed by rising valley

- shape_type: the type of F0 shape as defined in Table 2.1,
- tTilt_amp and tTilt_dur: the difference of rising and falling amplitudes and durations divided by their summation.

The hill shape in the conventional Tilt modeling can be represented in following equation:

$$F0(t) = A_{abs} + A - 2A(\frac{t}{D})^2 \qquad (0 < t < \frac{D}{2}) \tag{2.37}$$

$$F0(t) = A_{abs} + A - 2A(1 - \frac{t}{D})^2 \qquad (\frac{D}{2} < t < D) \tag{2.38}$$

where $F0(t)$ is the F0 value at a time t, A is a rising or falling amplitude, D is a rising or falling duration and A_{abs} is an absolute F0 value at the starting point of the rising or falling curve. In the T-Tilt modeling, for tonal languages, the vally shape can be represented in two more equations as follow:

$$F0(t) = A_{abs} + A - A(\frac{t}{D})^2 \qquad (0 < t < D) \tag{2.39}$$

$$F0(t) = A_{abs} + A(1 - \frac{t}{D})^2 \qquad (0 < t < D). \tag{2.40}$$

In the training stage, first, linguistic and acoustic information of training speech data is extracted. Each syllable of the training speech corpus is tagged with T-Tilt event labels. After that, regression trees (CART) are built separately for each T-Tilt parameter on the syllable-based event to predict each

of eight T-Tilt parameters. In the synthesis process converting T-Tilt parameters into an F0 contour, there are two steps including converting T-Tilt parameters to the rise/fall/connection description and converting to the F0 contour using Eqs. (2.37)–(2.40).

2.8.2 Spectral reproducibility

In the conventional HMM-based speech synthesis system, the speech parameter generation algorithm is utilized to generate the spectral and excitation parameter from HMMs to maximize their output probabilities under constraints between static and dynamic features. By taking account of constraints between the static and dynamic features, it can generate smooth speech parameter trajectories. However, the speech samples synthesized with the generated speech parameters often sound muffled. One of the factors causing the muffled sound is over-smoothing of the generated speech parameters. 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. The following techniques are used to reduce the effect of too much smoothness of generated spectra and enhance the speech quality [8].

2.8.2.1 Post-filtering

The simplest way of compensating over-smoothing is emphasizing the spectral structure by using a post-filter, which was originally developed for speech coding. The use of post-filtering techniques can reduce buzzy and muffled sound [36], [37]. However, too much post-filtering often introduces artificial sounds and degrades the similarity of synthesized speech to that uttered by the original speaker [38].

2.8.2.2 The utilization of multiple-level statistics

Another way of compensating over-smoothing is integrating multiple-level statistical models to generate speech parameter trajectories. One of the

most successful methods in this category is the speech parameter generation algorithm considering global variance (GV) of the mel-cepstral coefficients [13], [14] extracted from natural speech and those generated from HMM. Normally, the dynamic range of the generated mel-cepstral coefficients is smaller than that of the natural ones. The speech parameter generation algorithm considering GV has focused on solving this phenomenon. It tries to recover the dynamic range of generated trajectories close to those of the natural ones. GV, denoted by $\boldsymbol{v}_g$, is defined as an intra-utterance variance of a speech-parameter trajectory, c, as

$$\boldsymbol{v}_g = [v(1), ..., v(d), ..., v(D)]^\top \tag{2.41}$$

$$v(d) = \frac{1}{T}\sum_{t=1}^{T}\left(c_t(d) - \overline{c(d)}\right)^2 \tag{2.42}$$

$$\overline{c(d)} = \frac{1}{T}\sum_{t=1}^{T} c_t(d). \tag{2.43}$$

We calculate GVs for all training utterances and model them by using a single multi-variate Gaussian distribution as

$$P(\boldsymbol{v}_g|\boldsymbol{\lambda}_{GV}) = \mathcal{N}(\boldsymbol{v}_g; \boldsymbol{\mu}_g, \boldsymbol{U}_g) \tag{2.44}$$

where $\boldsymbol{\mu}_g$ is a mean vector and $\boldsymbol{U}_g$ is a covariance matrix of GVs. The speech parameter generation algorithm considering GV maximizes the following objective function with respect to c, i.e.,

$$\mathcal{L}_q^{(GV)} = \log P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda})^{\omega_g} + \log P(\boldsymbol{v}_g|\boldsymbol{\lambda}_{GV}) \tag{2.45}$$

where ω_g is a weight to balance the HMM and GV probabilities. The second term can be viewed as a penalty to prevent over-smoothing because it works to retain the dynamic range of the generated trajectory close to that of the training data. This method can be viewed as a statistical post-filtering technique to a certain extent.

To improve the GV algorithm, incorporating GV into the training part of HMM-based speech synthesis has also been proposed [39]. In this technique, Wu et al. introduced GV into the minimum generation error (MGE) training, where an additional generation error component measuring the distortion

between the generated GV and original one is introduced in generation error definition, and the parameters of HMMs are optimized so as to minimize the new generation error function. In order to normalize the scale of the generation error components for static feature and GV of feature trajectory, we denote

$$\sigma(\boldsymbol{c}) = [\sigma(1), \sigma(2), ..., \sigma(d), ..., \sigma(D)] \tag{2.46}$$

$$\sigma(d) = \sqrt{v(d)} \tag{2.47}$$

and use $\sigma(\boldsymbol{c})$ instead of $\boldsymbol{v}_g$ to calculate the generation error. The Euclidean distance is also adopted to calculate the distortion between the GV of generated trajectory and that of original one.

$$D_v(\sigma(\hat{\boldsymbol{c}}_q), \sigma(\boldsymbol{c})) = \|\sigma(\hat{\boldsymbol{c}}_q) - \sigma(\boldsymbol{c})\|^2. \tag{2.48}$$

The GV distortion and the original feature distortion are combined and the new generation error for $\boldsymbol{c}$ is defined as

$$\acute{e}(\boldsymbol{c}, \sigma(\boldsymbol{c}), \boldsymbol{\lambda}) = D_c(\hat{\boldsymbol{c}}_q, \boldsymbol{c}) + wD_v(\sigma(\hat{\boldsymbol{c}}_q), \sigma(\boldsymbol{c})) \tag{2.49}$$

where w denotes the GV weight for controlling a balance between these two distortions.

The generalized probabilistic descent algorithm is applied to minimize the generation errors. For each training utterance c_n , the updating rule of the HMM parameters is

$$\boldsymbol{\lambda}(n+1) = \boldsymbol{\lambda}(n) - \frac{\partial \acute{e}(c_n, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}}\bigg|_{\boldsymbol{\lambda}=\boldsymbol{\lambda}(n)}. \tag{2.50}$$

Under the definition of generation error, we obtain

$$\frac{\partial \acute{e}(\boldsymbol{c}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} = \frac{\partial}{\partial \boldsymbol{\lambda}}[(\hat{\boldsymbol{c}} - \boldsymbol{c})^2 + (\sigma(\hat{\boldsymbol{c}}) - \sigma(\boldsymbol{c}))^2]. \tag{2.51}$$

For the mean parameter $\boldsymbol{\mu}$, the equation is written as

$$\frac{\partial \acute{e}(\boldsymbol{c}, \boldsymbol{\lambda})}{\partial \boldsymbol{\mu}} = \frac{\partial}{\partial \boldsymbol{\mu}}(\hat{\boldsymbol{c}} - \boldsymbol{c})^2 + \frac{\partial}{\partial \boldsymbol{\mu}}(\sigma(\hat{\boldsymbol{c}}) - \sigma(\boldsymbol{c}))^2. \tag{2.52}$$

Finally, the updating rule for the mean vector is

$$\boldsymbol{\mu}(n+1) = \boldsymbol{\mu}(n) - 2\boldsymbol{\zeta} R^{-1} W^\top U^{-1} \tag{2.53}$$

where

$$\boldsymbol{\zeta} = (\hat{\boldsymbol{c}} - \boldsymbol{c})^\top + \frac{2}{T}(\sigma(\hat{\boldsymbol{c}}_q) - \sigma(\boldsymbol{c}))(\hat{\boldsymbol{c}} - m(\hat{\boldsymbol{c}})). \tag{2.54}$$

For the variance parameters, the gradient of generation error function is calculated as

$$\frac{\partial \acute{e}(\boldsymbol{c}, \boldsymbol{\lambda})}{\partial \boldsymbol{U}} = 2\boldsymbol{\zeta} R^{-1} W^\top (\boldsymbol{\mu} - W\hat{\boldsymbol{c}}). \tag{2.55}$$

The computational complexity of MGE-GV training is similar to that of MGE training, since the most computational cost in parameter updating is still related to the calculation of R^{-1}.

2.9 Conclusion

In this chapter, the basic statistical parametric speech synthesis system based on HMM framework was described. The MSD-HMM was proposed for F0 pattern modeling and generation. By using HSMM, the duration features of speech units can be appropriately modeled. The decision tree-based context clustering technique is used to cluster HMM states and share model parameters among states in each cluster. The speech parameters of synthetic speech is generated fully statistically based on an ML criterion. The model adaptation technique using average voice model was also described. By using model adaptation, the model training can be possible with only a small amount of training data of the target speaker. We have also reviewed the drawbacks of the conventional statistical parametric speech synthesis technique in two issues of tone correctness in tonal language and the spectral reproducibility. These issues are investigated in Chapter 3 and Chapter 4, respectively.

Chapter 3

Tone-Modeling Using a Quantized F0 Context in Average-Voice-Based Speech Synthesis

This chapter describes a technique for improving tone correctness in speech synthesis of a tonal language based on an average-voice model trained with a corpus from nonprofessional speakers' speech. We focus on reducing tone disagreements in speech data acquired from nonprofessional speakers without manually modifying the labels. To reduce the distortion in tone caused by inconsistent tonal labeling, quantized F0 symbols are utilized as the context for F0 to obtain an appropriate F0 model. With this technique, the tonal context label can be directly extracted from the original speech and this prevent inconsistency between speech data and F0 labels generated from transcriptions, which affect naturalness and the tone correctness in synthetic speech. We examine two types of labeling for the tonal context using phone-based and sub-phone-based quantized F0 symbols. Subjective and objective evaluations of the synthetic voice are carried out in terms of the intelligibility of tone and its naturalness. The experimental results from both the objective and subjective tests show that the proposed technique can improve not only naturalness but also the tone correctness of synthetic speech under conditions

where a small amount of speech data from nonprofessional target speakers is used.

3.1 Introduction

The recent development of corpus-based speech synthesis has greatly improved the quality and naturalness of synthetic speech in text-to-speech (TTS) systems. This improvement was mainly brought about by concatenative synthesis based on unit selection and statistical parametric synthesis based on hidden Markov models (HMMs) [7]. When we use speech data under good conditions, i.e., a sufficient amount of speech data uttered by a professional narrator and carefully labeled with manual modifications, the quality and naturalness of synthetic speech are satisfactory. However, a more flexible system that can synthesize speech using only a small amount of automatically labeled speech data from nonprofessional speakers is required to widen the applications of speech synthesis [40].

When we directly apply the current systems under such under-resourced conditions, we encounter some undesirable problems. One of the most serious of these is the deteriorated accuracy in labeling speech data. This is not a difficult task in labeling phonemes when the speech samples are uttered in reading style and their transcriptions are known. In contrast, there are sometimes disagreements between the prosodic labels and recorded speech samples when the labels are automatically created from transcriptions. The accuracy of labeling tonal information, especially in tonal languages, plays a primary role in the quality of the resulting synthetic speech. This is because incorrect tonal labels strongly affect the tone correctness of synthetic speech and such tone distortion can degrade not only the naturalness but also the intelligibility of speech. However, it is not always an easy task to manually modify the tonal labels at low cost. In such cases, we are forced to use data from nonprofessional speakers without making manual modifications.

A variety of techniques have been proposed to improve the tone correctness of synthetic speech in tonal languages. When a large amount of speech data from a target speaker is available, a concatenative approach with unit selection of a fundamental frequency (F0) contour provides good

performance [41], [42]. However, these F0 modeling approaches suffer from insufficient amounts of speech data when they are applied to under-resourced languages. HMM-based speech synthesis is one of the most attractive approaches to relaxing the data sparseness problem in unit-selection-based synthesis. Tone correctness in HMM-based speech synthesis has been improved by designing decision tree structures for Thai [43]. The system that was constructed could produce the prosody of synthetic speech better than a unit-selection-based TTS system [12] in terms of naturalness and tone correctness.

Introducing generative F0 models is another approach to avoiding the problem with insufficient amounts of training data. T-Tilt model [3], which is an expansion of Tilt model [35], was proposed to model the F0 contours of tonal languages. Fujisaki model [44] is also a well-known and effective F0 modeling technique and the model parameters were incorporated into average-voice-based speech synthesis [45]. These F0 modeling approaches represent an F0 contour with a set of model parameters, and these parameters are estimated from speech corpora using machine-learning methods such as the classification and regression tree [46]. However, since these techniques employed tonal information determined from transcriptions, their performance has still depended on the accuracy of tonal labeling. Consequently, inconsistency between speech data and tonal context label has arisen in automatic labeling processes in such situations, and tone distortion has appeared in synthetic speech.

We focus on the speech synthesis of Thai in this chapter, which is a tonal language, and propose an alternative technique of improving the tone correctness of synthetic speech for average-voice-based speech synthesis. We use quantized F0 symbols [47] as the context for F0 at a segmental level to reduce the tonal distortion caused by inconsistent tonal labeling. The quantized F0 symbols are directly determined from speech data and thus there are no inconsistencies between the speech data and F0 context labels. We examine two types of tonal context labeling. The first is an approach using phone-based F0 symbols, which was originally proposed for unsupervised F0 context labeling in pitch accent languages. The second is using sub-phone-based F0 symbols, which are determined for smaller units than phones. The

sub-phone-based F0 symbols are expected to be suitable to represent the tonal curve of an F0 contour more precisely than phone-based ones.

We need to automatically create context labels from a given text in the synthesis stage. Not only the results of text analysis but also an F0 symbol sequence representing correct tones is required to achieve this. To predict a feasible F0 context for an unseen text, we use a synthetic F0 contour generated from a *reference model*—a speaker-dependent model that is trained using a professional speaker's speech with conventional context labels that can generate appropriate F0 contours that represented correct tones. We compare the performance of the tonal context labeling based on the quantized F0 context and the conventional one based on transcriptions through both objective and subjective evaluations.

The structure of this chapter is as follows. The next section introduces the structure of Thai syllables and discusses the labeling problem in a non-professional multi-speaker corpus. Section 3.3 explains how we reduced tone disagreement by using two types of quantized F0 symbols, i.e., phone-based and sub-phone-based F0 symbols. The generation of context labels for speech synthesis is described in Section 3.4. Section 3.5 provides an overview of the proposed system including two main subsections on model training and speech synthesis. The experiments and discussions are presented in Section 3.6. Conclusions are given at the end of this chapter.

3.2 Labeling problem in HMM-based Thai speech synthesis

3.2.1 Structure of Thai syllables

The structure of Thai syllables is often described in the form of [C_i−V−C_f−T] or [C_i−V−T], where C_i denotes the initial consonant (including single and double consonants), V denotes the vowel (both short and long vowels), C_f denotes the final consonant (some single consonants), and T denotes the tone. Each syllable offers a choice from five distinct tones. Tones are often named according to their F0 shapes and levels such as medium

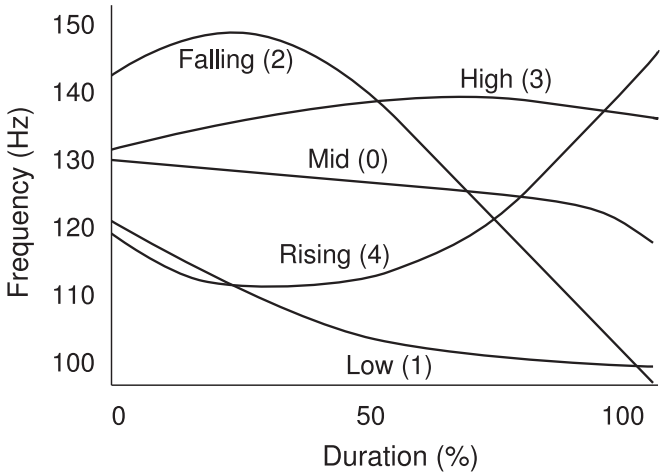


Figure 3.1: Typical syllable-unit F0 contours in Thai.

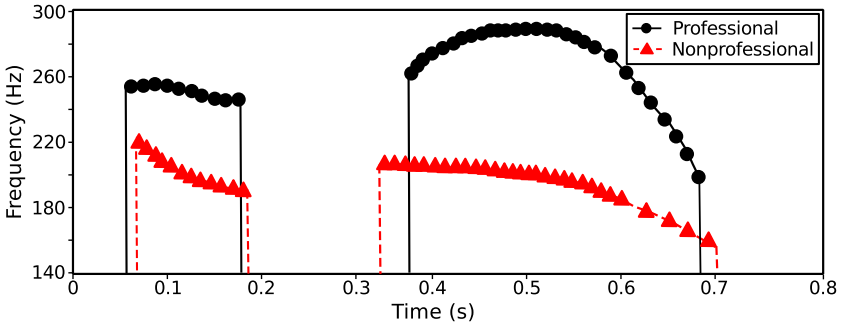


Figure 3.2: Example of F0 contours of natural speech uttered by professional and nonprofessional speakers.

(tone 0), low (tone 1), falling (tone 2), high (tone 3), and rising (tone 4). Figure 3.1 illustrates typical syllable-unit F0 contours for Thai. Note that each tone can be distinguished by the F0 trajectory at the syllable level and this affects the word meaning.

3.2.2 Labeling problem in tonal languages

The tonal context label automatically generated by text analysis is commonly used in the model training stage of conventional HMM-based speech synthesis of tonal languages. For instance, a Thai word [s-ua-jˆ] having a rising tone (tone 4), means “beautiful”, is labeled as [s-ua-jˆ-4], where “4” rep-

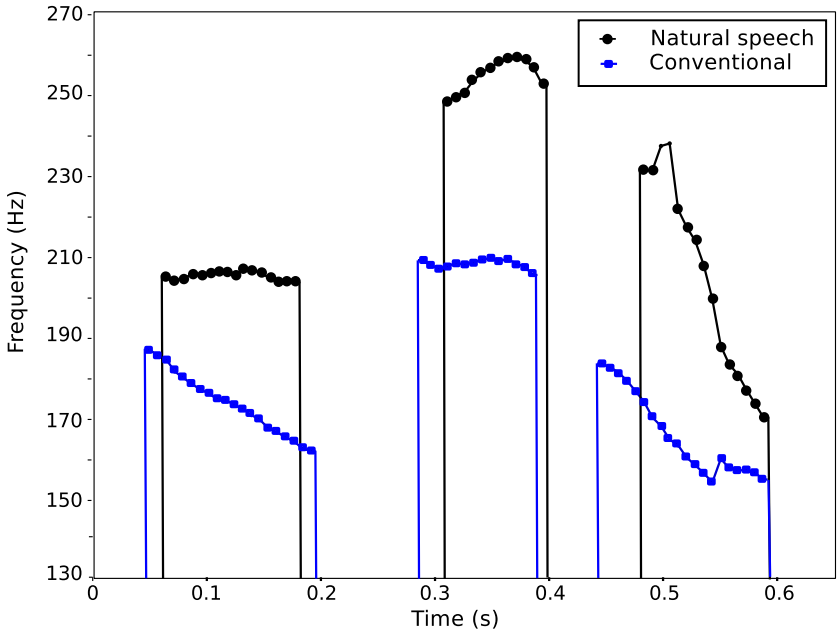


Figure 3.3: Comparison of F0 contours of natural and synthetic speech samples using conventional technique.

resents the rising tone type. Tonal context labeling generally only depends on the transcription of speech data. Therefore, when the target speaker is a nonprofessional speaker who has little experience with speech recording, he/she sometimes utters speech in incorrect tones and this leads to disagreements between speech data and tone labels. Such low-conditioned training data in model training affect the reliability of tone correctness in the acoustic model. As a result, the generated F0 contour sometimes has incorrect tones, which degrade the naturalness and intelligibility of synthetic speech. Figure 3.2 shows examples of the F0 contours of a Thai word /t-uːa-0 p^h-uː-2/, meaning "male" in English, uttered by professional and nonprofessional speakers. Here, the nonprofessional speaker has uttered this word in an incorrect tone. More specifically, the speaker has uttered the syllable /p^h-uː/ with tone "0", representing the mid tone type, but the tone of this syllable in the transcription from text analysis is "2", representing the falling tone type. This example demonstrates the inconsistency between speech data and transcriptions.

Tone disagreement is a crucial problem with corruption in tone when average-voice speech is generated by an HMM-based speech synthesis system with conventional contextual labeling. In other words, the tone correctness of synthetic speech is considerably degraded. This problem obviously emerges in the system where a nonprofessional multi-speaker speech database is used for training. Figure 3.3 compares the F0 contour of natural speech and that of synthetic speech generated by the conventional training system [31] for the Thai sentence /m-iː-ʔˆ-0 kʰ-uː-ʔˆ-2 kʰ-x-ŋˆ-1/, meaning "There are competitors..." in English. Tone distortion can be more easily observed in the F0 contour of the conventional approach than that in natural speech.

3.3 Reduction in tone disagreement for model training

To avoid the tone disagreement between speech data and tonal context labels, we should generate context labels from the speech data themselves and not from transcriptions. We employ quantized F0 symbols that had originally been proposed [47] for unsupervised F0 context labeling in pitch accent languages to achieve this. In addition, we arrange the conventional F0 context for tonal languages.

3.3.1 Quantized F0 symbols for tonal labeling

Let us assume that the log F0 values of the training data follow a normal distribution. When calculating the quantized F0 symbols, we first normalize the distribution of the log F0 sequence into the standard normal distribution $\mathcal{N}(0,1)$ for each utterance using the global mean and variance of log F0, which are obtained from the input speech in advance. Then, the F0 symbol s_p for each phone unit p is obtained by quantizing the mean value of log F0, f_p, of each phone into a discrete value as:

$$s_p = Q[f_p], \qquad s_p \in \{0, 1, \ldots, M-1\} \tag{3.1}$$

where $Q[.]$ denotes the operation of scalar quantization and M is the number of quantization levels. We set the points that equally divide the region $[-2, 2]$

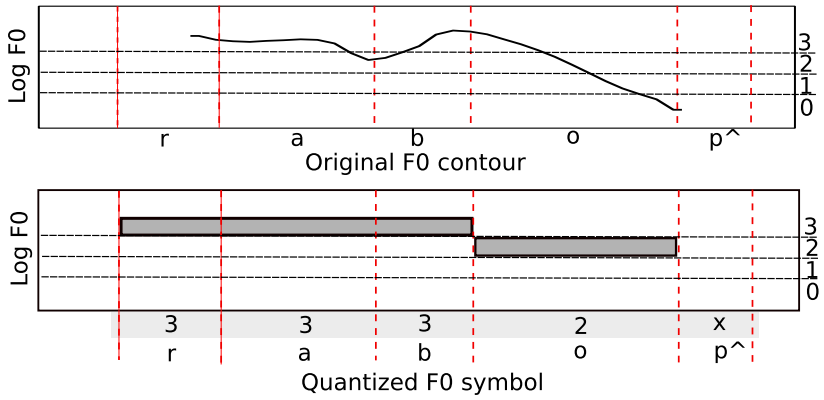


Figure 3.4: Example of phone-based F0 symbols.

into M intervals as the quantization boundaries. We examine two approaches in this study to obtain the tonal context labels, i.e., phone-based and sub-phone-based quantized F0 symbols.

3.3.2 Phone-based F0 symbols

The first approach is done in the same way as that proposed by [47], which will be referred to as phone-based F0 symbols. Each quantized F0 symbol in an utterance in this technique is labeled by using the phone boundary information in the time domain obtained from the transcription. The tonal context labels are obtained using F0 normalization and quantization in the way described in Section 3.3.1. Figure 3.4 illustrates an example of four-level quantized F0 labeling based on the phone boundary information. From the figure, the phone-based F0 symbol sequence of a Thai word /r-a-3 b-o-pˆ-1/ having a high tone (tone 3) and a low tone (tone 1), meaning "system" in English, is /3-3 3-2-x/ (the "-" symbol is used to separate the F0 symbols of each phoneme).

3.3.3 Sub-phone-based F0 symbols

As seen in Figure 3.4, the generated phone-based F0 symbol sequence is insufficient to represent correct information on tone contours. To solve this

problem, we propose sub-phone-based F0 symbols where we quantize the F0 contour based on units smaller than phones. A straightforward way of determining a segment is to use the state boundaries obtained by HMM-based forced alignment. However, we do not adopt a state-based segment because the most dominant factor in forced alignment is spectral characteristics and these result in boundaries that are inconsistent with the characteristics of an F0 sequence. Therefore, to alleviate the problem with inappropriately portioned state boundaries caused by inconsistency of characteristics in state segmentation, we equally divide each phone duration interval into portions in such a way that the number of portions is equal to that of the states of each phone HMM. The quantized F0 symbols are then calculated in all sub-phone-based portions and used as their labels.

Figure 3.5 shows an example of four-level quantized F0 labeling based on sub-phone boundary information. From the figure, the sub-phone-based F0 symbol sequence of a Thai word /r-a-3 b-o-pˆ-1/ having a high tone (tone 3) and a low tone (tone 1) is /“x_x_x_3_3”-“3_3_3_3_2” “2_2_3_3_3”-“3_3_2_1_0”-“x_x_x_x_x”/ (the “-” symbol is used to separate the F0 symbols of each phoneme and the “_” symbol is used to separate the F0 symbol of each sub-phoneme). Each sub-phone F0 symbol corresponds to the context of each state. For instance, the F0 symbol set of the phoneme /o/ from the syllable /b-o-pˆ-1/ is “3_3_2_1_0”. Therefore, in the third state of the HMM, the third F0 symbol “2” is utilized as the context for the third state.

3.4 Generation of context labels for speech synthesis

The phonetic and tonal context label sequence for synthesis in conventional HMM-based speech synthesis systems is obtained by using text analysis from the input text. However, the F0 context in the proposed technique cannot be directly generated from the input text because F0 values cannot be determined from only the text. To overcome this problem, we utilize F0 sequences generated from a reference model. We use a professional speaker for the reference model because the F0 context label sequence that is generated must

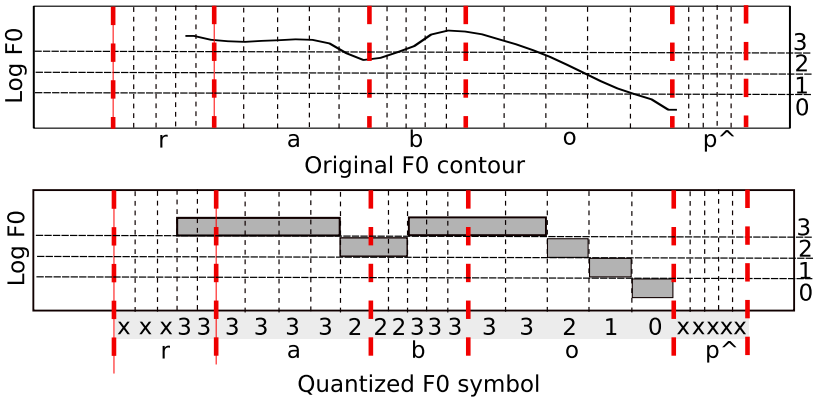


Figure 3.5: Example of sub-phone-based F0 symbols.

represent correct tone shapes. Note that conventional context labels are used in this process. We evaluated tone intelligibility and the naturalness of synthetic speech generated from the reference model to clarify the effects of the reference model, which is discussed in Section 3.6.3.

We first train the reference model with professional speech data using conventional context labels. The F0 sequence is then generated from the parameters obtained from the given input text in an ordinary HMM-based speech synthesizing procedure. The tonal context labels are obtained from the generated F0 sequence using F0 normalization and canalization as described in Section 3.3.1. We need to know the mean and variance parameters of the log F0 distribution of synthetic speech in F0 normalization. To do this, we generate the F0 sequences for all training sentences in advance, and calculate the global mean and variance parameters.

3.5 System overview

Figure 3.6 shows a block diagram of the proposed TTS system. To synthesize speech while only using a small amount of training data, we employ a framework of average-voice-based speech synthesis [48], [49] to train the model for the target speaker.

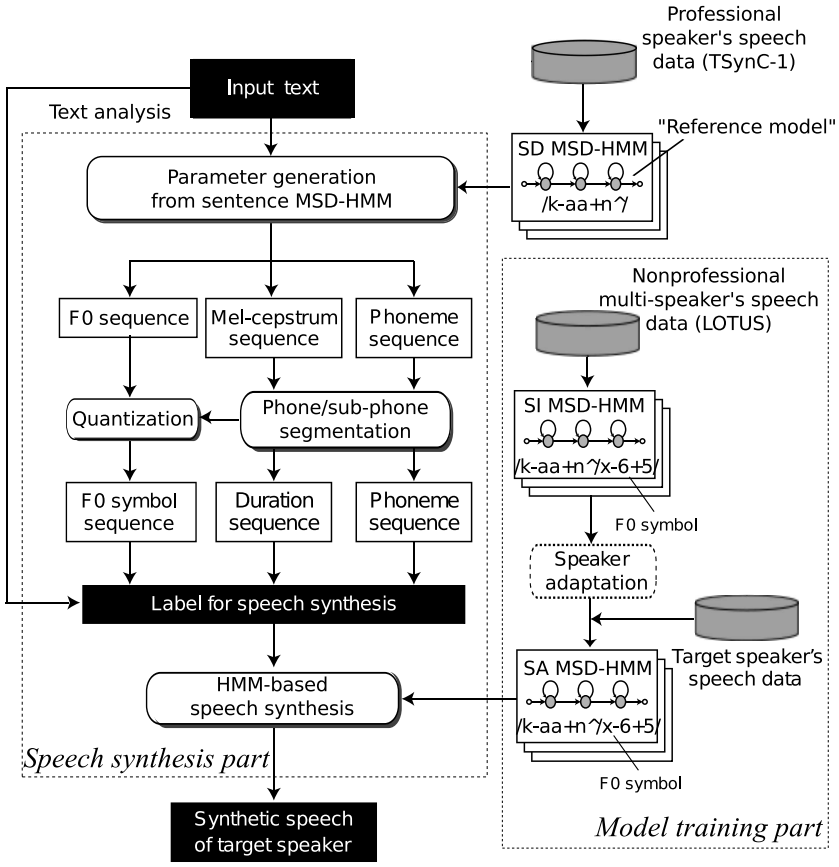


Figure 3.6: Overview of proposed TTS system.

3.5.1 Model training

First, an average voice model is trained using multiple data from nonprofessional speakers' speech. The tonal labeling described in Section 3.3 is used to improve the accuracy of F0 modeling for the average voice model. The spectrum and F0 are modeled with multi-stream HMMs in which the output distributions for the spectral and F0 parts are modeled using a continuous probability distribution for the former and a multi-space probability distribution [17] for the latter. Duration is modeled with a hidden semi-Markov model (HSMM) [18], which has an explicit duration distribution. In this study, we will use a speech corpus that contained a large number of speakers (48 speakers) but a small amount of data for each speaker ($\approx$ 5.2 minutes per

speaker). The speech corpus details are described in Section 3.6.1. Respective nodes in the decision tree of the average-voice model do not always have training data for all speakers in the speech database and some nodes could have data from only one speaker. Such speaker-biased nodes degrade the quality of both the average voice and synthetic speech after speaker adaptation is done. Therefore, we use shared decision tree context clustering (STC) [30] for tying model parameters, where every node in the decision tree always has the training data of all speakers so that each distribution of the average voice model reflects the statistics of all speakers. We also use speaker adaptive training in the procedure to train the average voice model [31] to improve its quality.

The adaptation data in the adaptation process uttered by a nonprofessional speaker are also labeled with the phonetic and F0 contexts in the same way as the training data for the average voice model. Then, the average voice model is adapted to the target speaker using a speaker adaptation algorithm. We use a combined method of maximum likelihood linear regression and maximum a posteriors adaptation based on HSMM [50], [51] in this study.

3.5.2 Speech synthesis

We prepare a reference model of a professional speaker to generate the proposed labels. When an input text is given, the conventional context labels are automatically generated from the given text by text analysis and a synthetic F0 contour is generated from the reference model using conventional labels. The F0 context labels are created using the F0 magnetization described in Section 3.3. The context labels to synthesize the target speaker's speech are created using the F0 context labels and the conventional labeling except the tonal context labels obtained from the given text via text analysis. In other words, we employ the F0 context labels in place of the conventional tonal context labels. Then, the spectral and F0 feature sequences are generated using the HMM-based parameter generation algorithm [25] and a speech waveform is synthesized using a speech synthesis filter.

3.6 Experiments

3.6.1 Experimental conditions

A set of phonetically balanced sentences from the Thai-speech database called LOTUS [52] from NECTEC was used in the training, adaptation, and evaluation. A set of phonetically balanced sentences from the Thai-speech database called TSynC-1 [53] from NECTEC was used to train the reference model. Whole text sentences from both databases were collected from the Thai part-of-speech tagged ORCHID corpus [54]. The speech data in LOTUS were uttered by 24 female and 24 male nonprofessional speakers with a standard Thai accent, while the speech in TSynC-1 was uttered by a professional female speaker. To calculate the tone error rates in the training data, we randomly selected a subset of 20 utterances from the nonprofessional (LOTUS) speaker corpus. The average tone error rates of nonprofessional speaker training data is 10.28%. Phoneme labels were included in both databases and other linguistic and syntactic information, such as part-of-speech, from ORCHID was used to construct context-dependent labels with 79 different phonemes including silence and pause.

The speech signals were sampled at a rate of 16 kHz and the spectral features were then extracted by STRAIGHT analysis [55], [56] with a 5-ms shift. A feature vector consisted of 25 mel-cepstral coefficients including the zeroth coefficient, the log F0, and their delta and delta-delta coefficients. We used five-state left-to-right HSMM for the acoustic model. We used preceding, current, and succeeding F0 symbols for the quantized F0 context.

In the following experiments, we constructed three average-voice models and a reference model. Each average-voice model was trained using 35 sentences by each speaker, from 24 female and 24 male speakers of the LOTUS speech data, i.e., there were a total of 1680 training utterances. The three average-voice models were constructed using different tonal context labels, i.e., conventional, phone-based F0 symbol and sub-phone-based F0 symbol. The reference model was trained using 2500 sentences from the TSynC-1 speech data uttered by a professional speaker, where the conventional tonal context labels were used. The numbers of leaf nodes in the resultant decision

Table 3.1: Number of leaf nodes in the decision tree.

Type of model	Type of tonal context label	Mel-ceptrum	Log F0
Average-voice	Conventional	121	222
	Phone-based	115	208
	Sub-phone-based	117	219
Reference	Conventional	4157	11804

trees for mel-cepstrum and log F0 in each model are shown in Table 3.1. From the table, we see that the numbers of leaf nodes in the average voice models are much smaller than that in the reference model. This is because the amount of the training data for each speaker of the average voice models was much smaller than that for the reference model. In other words, there were a relatively small number of common questions applicable to all training speakers and this prevented the tree from growing down in the STC-based context clustering.

We used 35 sentences for each target speaker for adaptation and evaluation and chose two females (F1, F2) and two males (M1, M2) from the non-professional speech data (LOTUS) as the target speakers. We used the technique of leave-one-out cross-validation for evaluation to select the data for adaptation and evaluation. We evaluated three types of synthetic speech using conventional and proposed techniques. The first was synthetic speech with the conventional technique, where the conventional tonal context labels were used. The synthetic speech was generated from the adapted model where the same model adaptation algorithm as the proposed system was used. For the adaptation, the average voice model with the conventional context label was used. When the input text was given, all context labels were directly generated by text analysis. The second and third were those with the proposed techniques where phone-based and sub-phone-based F0 symbols were used, respectively, as the tonal context labels in the training of the average voice model and the speaker adaptation.

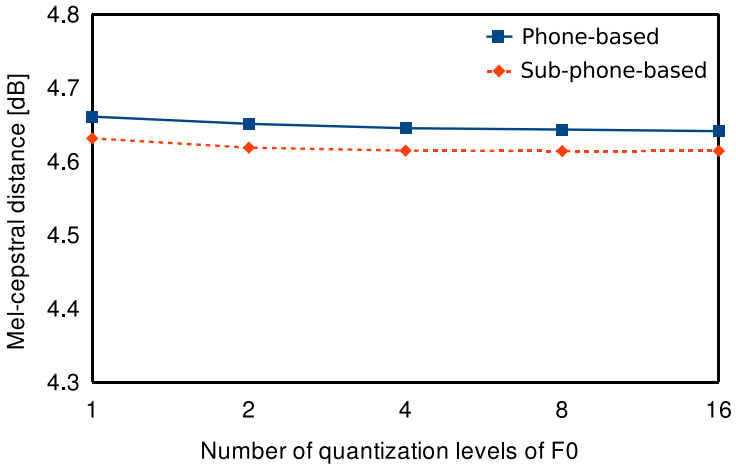


Figure 3.7: Average cepstral distances with different numbers of quantization levels.

3.6.2 Performance for different numbers of quantization levels

The number of quantization levels should be determined in advance when using the proposed tonal context labeling. We experimentally explored the appropriate number of quantization levels through objective evaluation. The phone-based and sub-phone-based F0 symbols were evaluated in terms of the mel-cepstral distance and RMS log F0 error when increasing the number of quantization levels from 1 to 16. The log F0 values were only evaluated in regions where both the generated and the original speech signals were voiced to calculate F0 distortion. The cepstral distances were calculated using the frames excluding the silence ones according to label information. We applied time alignment to the synthetic speech signals with the original utterances of the speaker to calculate both errors.

Figure 3.7 plots the average cepstral distance between original and synthetic speech with different numbers of quantization levels of 1, 2, 4, 8, and 16 for the tonal context label with phone-based and sub-phone-based F0 symbols. The experimental results indicate that both the phone-based and sub-phone-based F0 symbols have no significant differences for any number of the quantization levels. The average RMS error of log F0 with the different

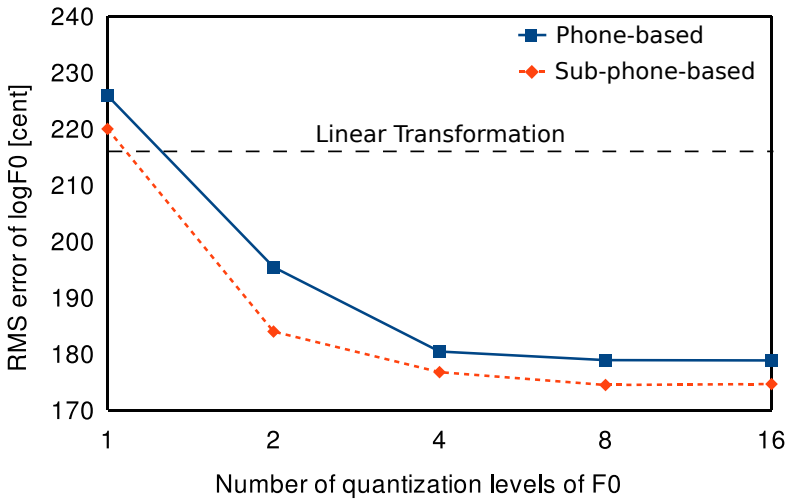


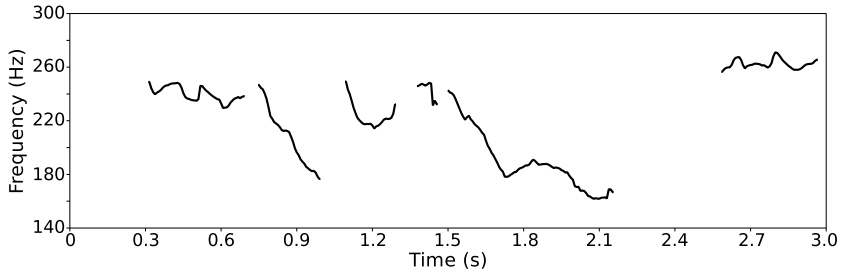
Figure 3.8: Average RMS errors with different numbers of quantization levels.

numbers of quantization levels of 1, 2, 4, 8, and 16 for the tonal context label with phone-based and sub-phone-based F0 symbols are plotted in Figure 3.8.

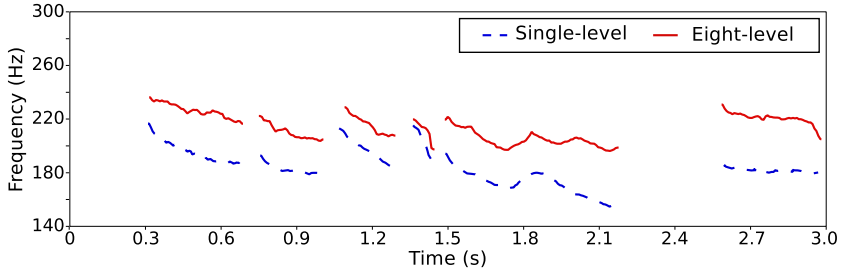
To examine the effectiveness of F0 modeling with the proposed F0 contexts, we also evaluated simple F0 conversion from the generated F0 of the reference model. In the conversion, we used a global linear transformation, which is widely used in voice conversion, given by

$$\hat{f}_y = \frac{f_x - \mu_x}{\sigma_x} \cdot \sigma_y + \mu_y, \tag{3.2}$$

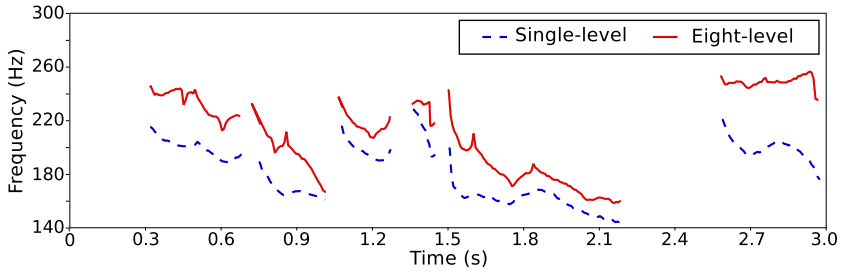
where f_x and $\hat{f}_y$ are the synthetic log F0 values of the reference speaker and the converted F0 to the target speaker. μ_x and σ_x are the mean and standard deviation calculated from the reference speaker's training data, and μ_y and σ_y are those from the target speaker's, respectively. The experimental results shown in Figure 3.8 reveal that F0 distortion decreases as the numbers of quantization levels increases in both phone-based and sub-phone-based F0 symbols. In addition, F0 distortion does not change significantly between 8- and 16-level quantization. The sub-phone-based F0 symbols outperform tonal context label with the phone-based F0 symbols at all numbers of quantization levels. Moreover, the F0 symbol-based methods outperformed the linear transformation when the number of quantization levels was more than one. Taking these results into consideration, we fixed the number of quanti-



(a) Original speech



(b) Single- and eight-level phone-based F0 symbols



(c) Single- and eight-level sub-phone-based F0 symbols

Figure 3.9: Examples of F0 contours for target speaker's original and synthetic speech.

zation levels to eight for both tonal context label with the phone-based and sub-phone-based F0 symbols in all experiments discussed in this chapter.

To clarify the effect of the choice of the number of quantization levels, examples of F0 contours are shown in Figure 3.9: (a) target speaker's natural speech, (b) synthetic speech samples using single- and eight-level phone-based F0 symbols, and (c) those using single- and eight-level sub-phone-based F0 symbols. The results show that the sub-phone-based F0 symbols can represent the F0 contour more appropriately than the phone-based F0 symbols.

3.6.3 Comparison of performance with conventional technique

3.6.3.1 Objective evaluation results

An objective evaluation was done by calculating distortion in the generated mel-cepstrum and log F0 of synthetic speech against those of the original speech. We compared three types of synthetic speech in the evaluation with different context labels as described in Section 3.6.1. We used the same procedure as that explained in Section 3.6.2 to calculate error. Figures 3.10 and 3.11 show the average cepstral distance and the average RMS errors with standard deviation in the generated log F0 and that extracted from the real utterances of the target speakers. For comparison, we also calculated the F0 distortion when using the linear transformation given by equation (3.2). The average RMS error of log F0 with the proposed tonal context labeling (sub-phone-based) is significantly smaller than that with conventional context labeling. Comparing the phone-based and sub-phone-based F0 symbols, there is little difference either in the cepstral distance or the RMS error of log F0. We can also see that the F0 distortion was alleviated by the proposed tonal context labeling compared to the global linear transformation. We can see that the proposed F0 modeling using quantized F0 contexts are indispensable to take advantage of the reference model. In other words, the simple linear transformation is insufficient to convert the speaker individuality to that of the target speaker, and explicit F0 modeling is necessary.

Figure 3.12 shows natural and synthetic F0 contours from both F0 symbol-based techniques. We can see that the sub-phone-based F0 symbols represent the F0 curve better than the phone-based F0 symbols. The Thai word /th-ii-2/ (meaning "at" in English) consists of two phonemes, an unvoiced initial consonant and a vowel. The normal F0 contour in Thai tone, especially in a falling tone (tone 2), is usually a rising and then a falling curve. When using the phone-based F0 symbol at 8-level quantization, the F0 symbol sequence is /x-4/, which can only represent the falling curve. For the sub-phone-based F0 symbol, the F0 symbol sequence becomes /"x_x_x_x_x"-"6_7_5_2_x"/, which gives a rising and then a falling curve. We can see from these results that the sub-phone-based F0 symbols yield a better outcome

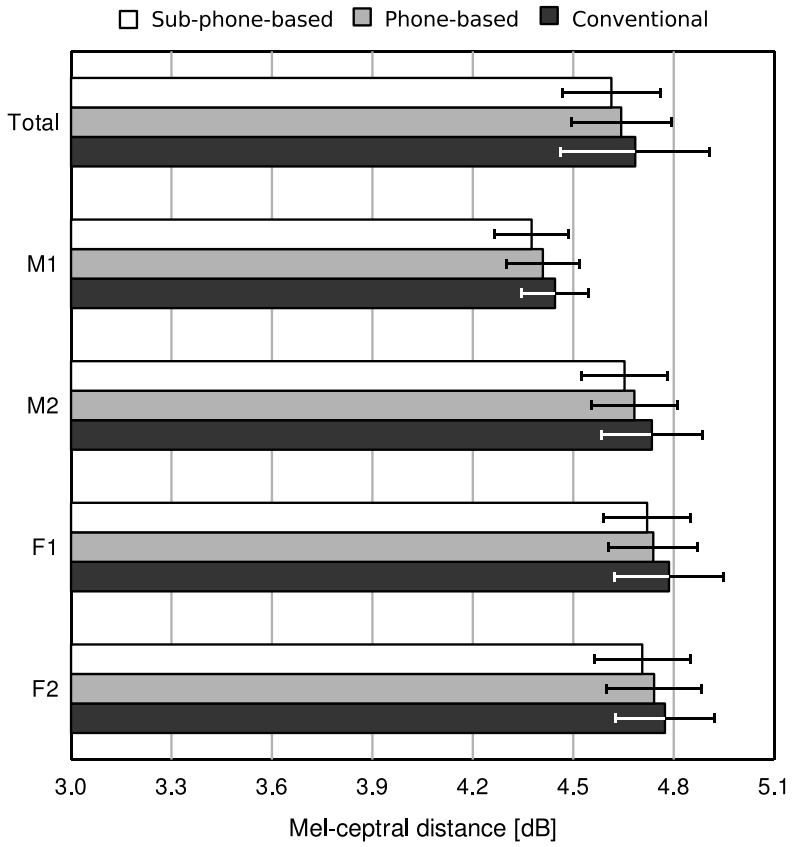


Figure 3.10: Comparison of average cepstral distances.

for the shape of the F0 contour in Thai tone.

The F0 contours generated from conventional contextual labeling, the tonal context label with phone-based F0 symbols, and the tonal context label with sub-phone-based F0 symbols have been compared with the F0 contours of natural speech in Figure 3.13. We can see from the results that the tonal context labeling we propose can generate an F0 contour closer to that of natural speech than the other techniques.

3.6.3.2 Subjective evaluation results

A human perceptual test was necessary to prove how effective the proposed technique was. We employed a mean opinion score (MOS) test to evaluate the perceptual quality in terms of naturalness and tone intelligibility. The

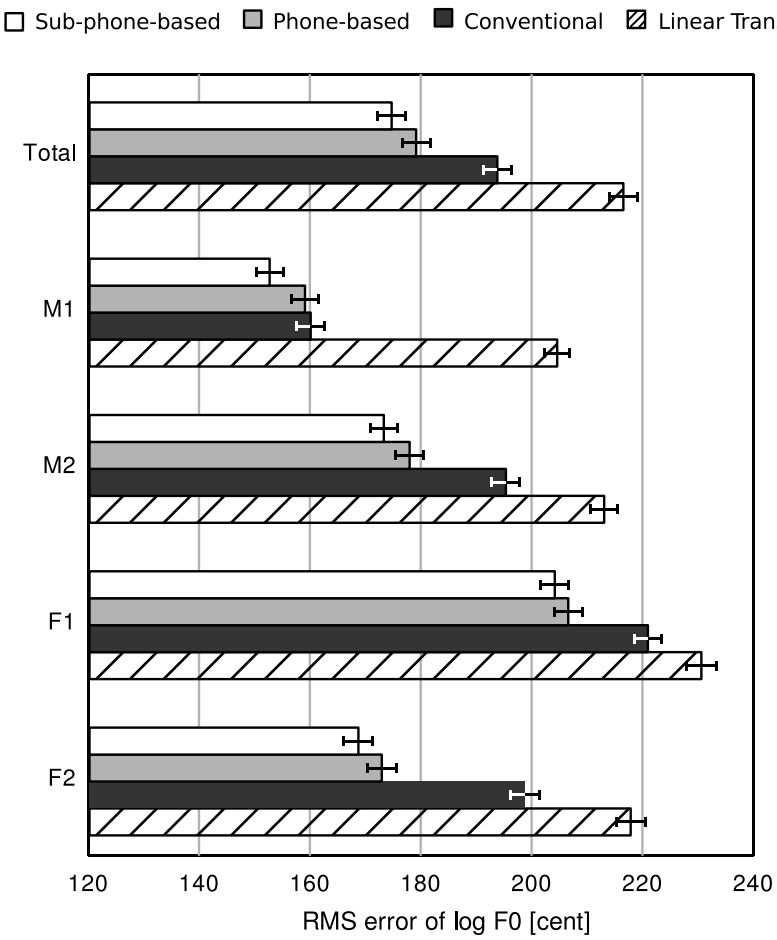


Figure 3.11: Comparison of average log F0 RMS errors.

20 utterances were randomly chosen as the test stimuli from the synthetic speech samples used in the objective test. It is noted that those utterances are excluded in the training set. We evaluated three types of synthetic speech from the average-voice-based system, which is presented in Section 3.6.3.1. We also evaluated the synthetic speech by using conventional context labels generated from the reference model. As a result, we compared four types of synthetic speech in the evaluation. Ten Thai native speakers listened and evaluated each utterance on a five-point scale from "1: bad" to "5: excellent" according to their satisfaction with the naturalness of tones perceived. Listeners could repeat sentences to evaluate any syllables as many times as they required to ensure that they were accurately evaluating syllables one-by-one.

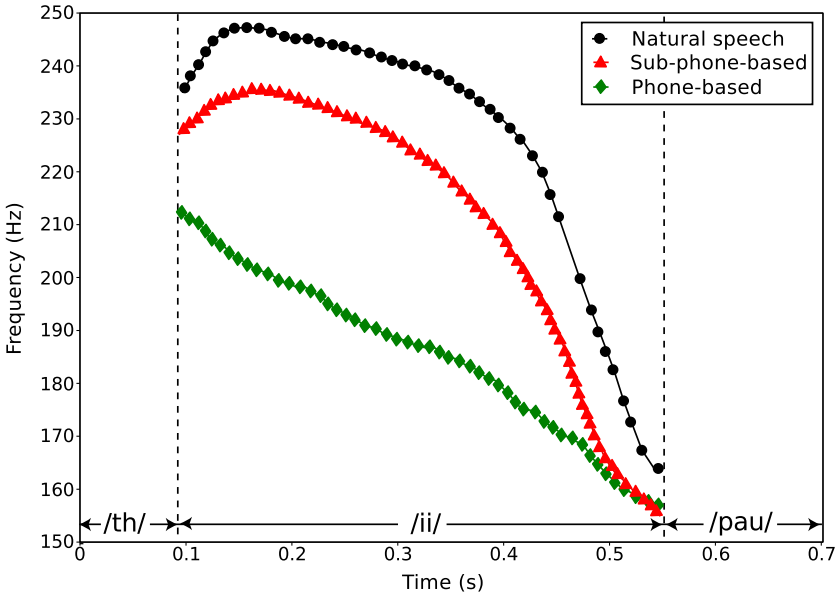


Figure 3.12: Natural and synthetic F0 contours of Thai word /th-ii-2/.

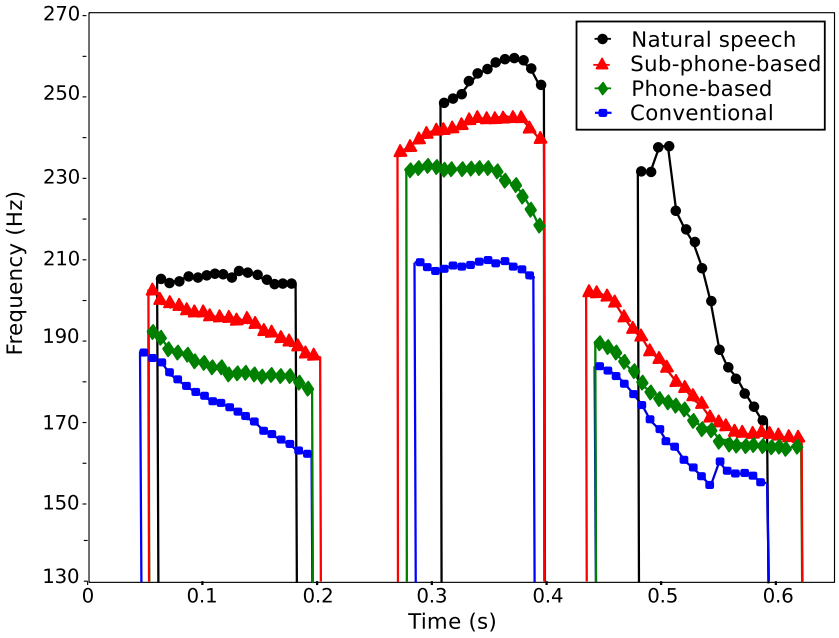


Figure 3.13: Comparison of F0 contours of natural speech and those generated from conventional contextual labeling, tonal context label with phone-based F0 symbols and tonal context label with sub-phone-based F0 symbols.

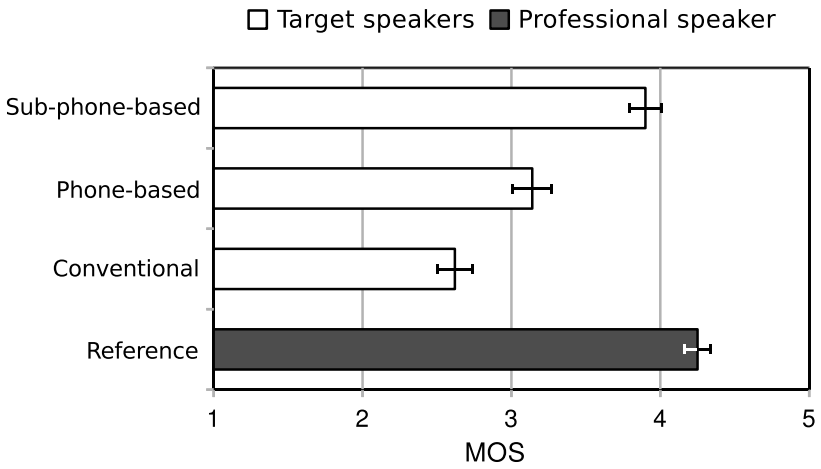


Figure 3.14: Results of MOS test on naturalness of tones perceived.

The scores in terms of naturalness with a confidence interval of 95% are given in Figure 3.14. We can see that context labeling with the sub-phone-based F0 symbols can significantly improve the naturalness of synthetic speech more than that with the other context labels for average-voice-based system.

Another subjective test was conducted on the basis of syllables rather than utterances to enable synthetic speech to be analyzed in terms of intelligibility. Listeners in this test were given the syllabic transcriptions of all test utterances. After they had listened to each test utterance, they were asked to mark syllables that had incorrect tones. The tone error rate (%) was the number of incorrect syllables divided by the total number of syllables in the test set. Figure 3.15 gives the average tone-error percentages for different context labels with a confidence interval of 95%. We can see that sub-phone-based F0 symbols can reduce the percentage of tone errors more than that with other context labels for the average-voice-based system.

3.6.3.3 Subjective evaluation results on the semantically unpredictable sentences

Generally, semantic and syntactic surrounding contexts often help understanding some unintelligible parts of synthetic speech and effect on intelligibility scores [57]. To avoid this effect on measuring the tone correctness of synthetic speech, semantically unpredictable sentences (SUS) were used

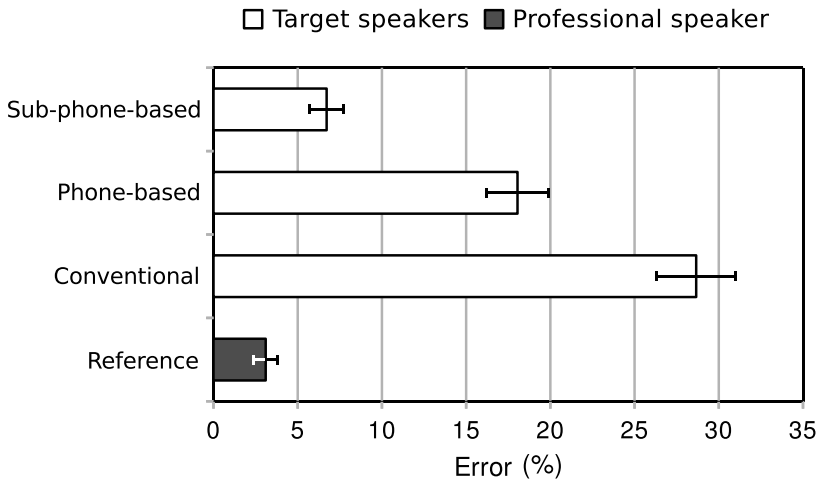


Figure 3.15: Percentage of tone errors in synthetic speech.

in subjective evaluation. In SUS test, the sentences are composed of words according to the syntactic structure of Thai language. For each sentence, we left one word blank in an answer sheet and let each listener fill in a word there. As a hint, we provided only the phonetic information of the missing word to enable the listeners to focus on the target word. Each word can be pronounced with various tones with different meanings and the listeners cannot predict it from the surrounding context. In other words, the listeners were unable to predict the tone of target word by only reading the sentence alone. It should be noted that we used the daily-life sentences in this experiment.

We compared four types of synthetic speech in the evaluation, which is presented in Section 3.6.3.2. Each test set contained 10 sentences of SUS. Twenty listeners who were native speakers of Thai took part in the experiments. In order to avoid learning effects, each participant listened to each sentence only once. Figure 3.16 gives the average tone-error percentages on the SUS with a confidence interval of 95%. From the results, it is seen that sub-phone-based F0 symbols can reduce the percentage of tone errors more than other context labels for average-voice-based system.

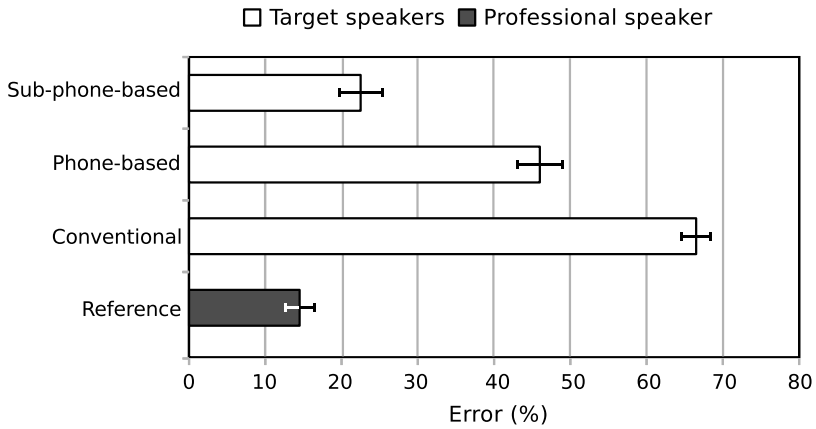


Figure 3.16: Percentage of tone errors on the SUS in synthetic speech.

3.7 Conclusions

We proposed a technique of modeling tones to improve the tone correctness of synthetic speech. Quantized F0 symbols in the proposed technique were utilized in two different ways based on phone and sub-phone boundary information. We found that F0 distortion decreased from the objective experimental results using tonal context label, phone-based and sub-phone-based F0 symbols more than that with the conventional tonal context labels. In addition, we found that the sub-phone-based F0 symbols represented the F0 curve better than the phone-based F0 symbols. The subjective tests also yielded results that corresponded to those from the objective tests. The experimental results from both the subjective and objective evaluations results confirmed that the tonal features we propose can alleviate the problem with tone disagreement. As a result, the tone correctness of synthesized speech was significantly improved. We intend to apply the proposed technique to expressive speech in future work especially within the context of tonal languages.

Chapter 4

Parameter Generation Using Local Variance for HMM-Based Speech Synthesis

This chapter describes a parameter generation algorithm using a local variance (LV) model in HMM-based speech synthesis. In the proposed technique, we define the LV as a feature that represents the local variation of a spectral parameter sequence and model LVs using HMMs. Context-dependent HMMs are used to capture the dependence of LV trajectories on phonetic and prosodic contexts. In addition, the dynamic features of LVs are taken into account as well as the static one to appropriately model the dynamic characteristics of LV trajectories. By introducing the LV model into the spectral parameter generation process, the proposed technique can impose a more precise variance constraint for each frame than the conventional technique with a global variance (GV) model. Consequently, the proposed technique alleviates the excessive spectral peak enhancement that often occurs in GV-based parameter generation. Objective evaluation results show that the proposed technique can generate better spectral parameter trajectories than the GV-based technique in terms of spectral and LV distortion. Moreover, the results of subjective evaluation demonstrate that the proposed technique can generate synthetic speech significantly closer to the original one than the conventional technique while maintaining speech naturalness.

4.1 Introduction

Parametric speech synthesis based on hidden Markov models (HMMs) is an effective framework for generating stable and diverse synthetic speech and is widely studied [8]. Specifically, this approach allows us not only to produce smooth and stable speech under a small footprint but also to add more variations to synthetic speech by using a variety of techniques: adaptation, interpolation, and control techniques for speaker characteristics, emotional expressions, speaking styles, and so on [50], [58]. In HMM-based speech synthesis, speech parameters, i.e., spectral and excitation features, and durations, of each speech synthesis unit are simultaneously modeled using context-dependent HMMs in a unified framework [7]. In the synthesis stage, a smooth speech parameter trajectory is generated by maximizing the output probability density function (pdf) determined by HMMs under a constraint between static and dynamic features [25].

However, one of the major problems in HMM-based speech synthesis is that the trajectories of spectral parameters generated from HMMs are often over-smoothed, and speech formants become unclear because of multiple factors, e.g., a parameter tying process during model training [59]. This causes the degradation of perceptual quality and makes synthetic speech sound buzzy and muffled. To alleviate this problem, a parameter generation algorithm considering the global variance (GV) [13] has been shown to be effective for improving the perceptual quality of synthetic speech. In this technique, GVs of spectral parameters, e.g., mel-cepstrum coefficients, are calculated for respective training utterances and are modeled by a single Gaussian pdf. This GV model is used for variance compensation in parameter generation, and a spectral parameter sequence is generated so as to maximize a weighted product of HMM and GV pdfs. Subjective evaluation results have demonstrated that variance compensation using a GV model significantly improves the naturalness of synthetic speech compared to a traditional parameter generation technique without GV modeling [14].

Recently, several techniques have been proposed for further improvement in the performance of GV-based variance compensation [60]–[63]. Context-dependent multiple GV models were used in [60], instead of a single GV

model, to capture the dependence of the GVs on sentence-level contextual factors such as the number of phonemes in each sentence. For more appropriate modeling of GVs, the log power spectrum of line spectrum pairs (LSPs) and frequency-domain delta LSPs were used instead of cepstral features in [61] and [62], respectively. When there is no computational limitation, introducing a framework of the product of experts [64] into HMM-based speech synthesis is also a promising approach [63] in which normalized Gaussian experts are used with the framework of trajectory HMMs [65].

There is also another approach where local variance (LV) was used instead of GV [39]. In this technique, an LV was determined for each frame of a spectral parameter sequence and was introduced into minimum generation error (MGE) training [66]. Since the GV is an utterance-level variance feature, it is difficult to capture the dependence on shorter linguistic units such as phonemes, syllables, and phrases. As a result, the spectral peaks of synthetic speech are sometimes excessively enhanced compared to those of natural speech, which degrades the similarity of the synthetic speech to the original one. In contrast, the LV is a frame-level feature and is capable of representing more precise variance characteristics than the GV. Although the experimental results in [39] did not show a clear advantage of using LV against the GV, LV-based variance compensation could have potential to outperform GV-based compensation.

In this work, we propose an alternative approach to utilizing LV-based variance compensation for HMM-based speech synthesis. Our approach differs from [39] in that the LV is taken into account not in the training stage but in the synthesis stage. To model the dynamic characteristics of LV trajectories, the proposed technique uses the dynamic features of LVs as well as the static one. Context-dependent HMMs are used to capture the dependence of LVs on phonetic and prosodic contexts. By using the context-dependent LV model, we can impose a more precise variance constraint on each frame in the parameter generation, which prevents excessive enhancement of dynamic characteristics in the generated spectral parameter sequence. In the experiments, first we examine an appropriate window size for LV calculation using a development set because the window size affects the accuracy of capturing the local variance characteristics of spectral parameter sequences. On

the other hand, a fixed window size of fifty frames, for which there was no discussion about the optimality, was used in [39]. Finally, we compare the performance of the proposed and conventional parameter generation techniques through both objective and subjective evaluation. In this work, we describe the detail of the similarity and difference of GV and LV. Moreover we evaluate the proposed technique using multiple target speakers and conduct subjective listening tests on both similarity and naturalness while only a single speaker was evaluated with a MOS test on similarity in [67].

4.2 Conventional parameter generation algorithm with GV model

The GV vector $\boldsymbol{v}_{\mathrm{g}} = [v(1), ..., v(d), ..., v(D)]^{\top}$ of a spectral parameter sequence $\boldsymbol{c}$ is calculated by

$$v(d) = \frac{1}{T}\sum_{t=1}^{T}\left(c_t(d) - \overline{c(d)}\right)^2 \tag{4.1}$$

$$\overline{c(d)} = \frac{1}{T}\sum_{t=1}^{T} c_t(d). \tag{4.2}$$

To consider the GV in the parameter generation process, the distribution of GVs is modeled by a single multivariate Gaussian pdf, given by

$$P(\boldsymbol{v}_{\mathrm{g}}|\boldsymbol{\lambda}_{\mathrm{GV}}) = \mathcal{N}(\boldsymbol{v}_{\mathrm{g}}; \boldsymbol{\mu}_{\mathrm{g}}, \boldsymbol{U}_{\mathrm{g}}) \tag{4.3}$$

$$= \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{U}_{\mathrm{g}}|}}\exp\left\{-\frac{1}{2}(\boldsymbol{v}_{\mathrm{g}} - \boldsymbol{\mu}_{\mathrm{g}})^{\top}\boldsymbol{U}_{\mathrm{g}}^{-1}(\boldsymbol{v}_{\mathrm{g}} - \boldsymbol{\mu}_{\mathrm{g}})\right\} \tag{4.4}$$

where $\boldsymbol{\lambda}_{\mathrm{GV}}$ denotes the GV parameter set including the mean vector $\boldsymbol{\mu}_{\mathrm{g}}$ and the covariance matrix $\boldsymbol{U}_{\mathrm{g}}$. In the training stage, the respective HMM and GV model parameter sets, $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}_{\mathrm{GV}}$, are independently estimated using the training data.

In parameter generation, the optimum parameter sequence $\hat{\boldsymbol{c}}$ is determined so as to maximize an objective function consisting of the HMM and

GV log pdfs as follows:

$$\mathcal{L}_{\boldsymbol{q}}^{(\mathrm{GV})} = \log P(\boldsymbol{o}|\boldsymbol{q},\boldsymbol{\lambda})^{\omega_{\mathrm{GV}}} + \log P(\boldsymbol{v}_{\mathrm{g}}|\boldsymbol{\lambda}_{\mathrm{GV}}) \tag{4.5}$$

$$\propto \omega_{\mathrm{GV}}(-\frac{1}{2}\boldsymbol{c}^{\top}\boldsymbol{W}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{W}\boldsymbol{c} + \boldsymbol{c}^{\top}\boldsymbol{W}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}}) + (-\frac{1}{2}\boldsymbol{v}_{\mathrm{g}}^{\top}\boldsymbol{U}_{\mathrm{g}}^{-1}\boldsymbol{v}_{\mathrm{g}} + \boldsymbol{v}_{\mathrm{g}}^{\top}\boldsymbol{U}_{\mathrm{g}}^{-1}\boldsymbol{\mu}_{\mathrm{g}}) \tag{4.6}$$

where the constant ω_{GV} denotes the GV weight for controlling the balance between the two pdfs. ω_{GV} is usually set to the ratio of the dimensions of vectors $\boldsymbol{v}_{\mathrm{g}}$ and $\boldsymbol{o}$, i.e., $\omega_{\mathrm{GV}} = 1/3T$ [14].

Although incorporating the GV pdf into the objective function alleviates the over-smoothing effect that is inevitable in the traditional parameter generation algorithm without variance compensation, the GV model represents only the utterance-level variance characteristics of spectral parameter sequences, and such rough variance compensation distorts synthetic speech in terms of a spectral distance measure.

4.3 Proposed parameter generation algorithm with LV model

4.3.1 Local variance with dynamic features

The LV vector, $\boldsymbol{v}$, of a spectral parameter sequence $\boldsymbol{c}$ is defined as

$$\boldsymbol{v} = [{\boldsymbol{v}_1}^{\top}, ..., {\boldsymbol{v}_t}^{\top}, ..., {\boldsymbol{v}_T}^{\top}]^{\top} \tag{4.7}$$

$$\boldsymbol{v}_t = [v_t(1), ..., v_t(d), ..., v_t(D)]^{\top} \tag{4.8}$$

$$v_t(d) = \frac{1}{L}\sum_{\tau=t-L_-}^{t+L_+}\left(c_\tau(d) - \overline{c(d)}_t\right)^2 \tag{4.9}$$

$$\overline{c(d)}_t = \frac{1}{L}\sum_{\tau=t-L_-}^{t+L_+} c_\tau(d) \tag{4.10}$$

$$L = L_- + L_+ + 1 \tag{4.11}$$

where L is the size of the window for the variance calculation [39]. It is noted that LV is calculated frame by frame. We set $t - L_-$ (the first frame number

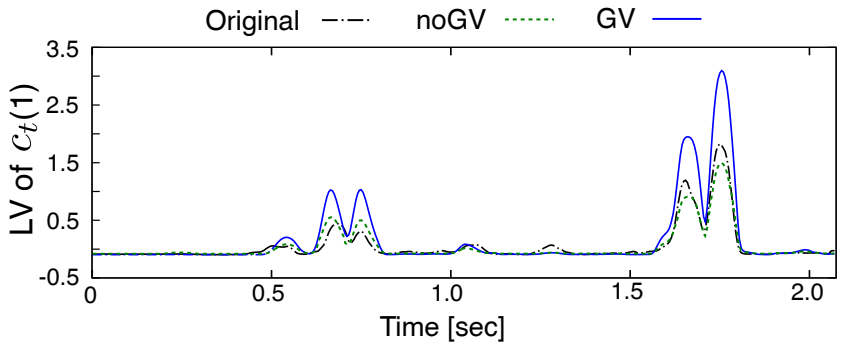
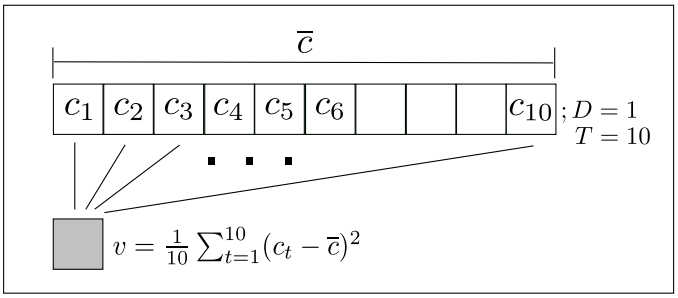


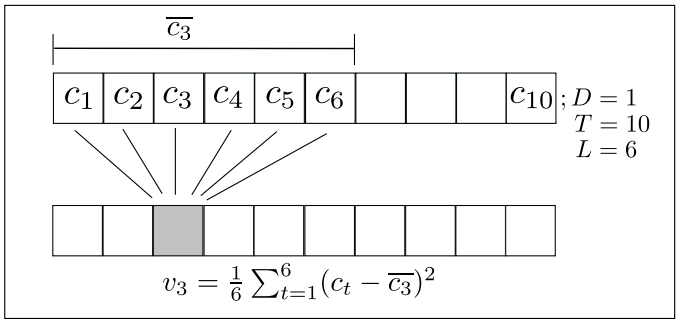
Figure 4.1: Example of LV trajectories calculated from the original and generated 1st mel-cepstral coefficient sequences.

of the window) to 1 when $t - L_-$ is below 1. In this case, L becomes $t + L_+$. Similarly, we set $t + L_+$ to T when $t + L_-$ is above T. In this case, L becomes $T - t + L_- + 1$. Fig. 4.1 shows an example of LV trajectories calculated from the 1st mel-cepstral coefficient sequences of original and synthetic speech. We set the window size to $L = 20 \quad (L_- = 9, L_+ = 10)$. "GV" and "noGV" represent the LV trajectories of synthetic speech with and without the GV model, respectively. From the figure, it is observed that the generated mel-cepstral sequence without GV sometimes gives LVs smaller than those of the original speech, which leads to degrading the naturalness of the synthetic speech. In contrast, although the LVs of the synthetic speech are increased by introducing the GV model, LV values in several regions become much larger than those in the original speech, and this implies the excessive enhancement of spectral peaks.

Fig. 4.2 shows a simplified example of GV and LV calculations assuming that $\boldsymbol{c}$ is a scalar sequence, i.e., $D = 1$, and the number of frames is 10. In the GV calculation of Fig. 4.2(a), a single GV value is calculated using all frames of $\boldsymbol{c}$. On the other hand, in the LV calculation of Fig. 4.2(b), each LV value is calculated using only the local parameters around the frame. For instance, suppose that the window size is five, we computed the local variance at $t = 3$ by using the frames c_1, c_2, c_3, c_4, and c_5. Consequently, LV has time-series values corresponding to the number of frames in an utterance. When the window size (L) in LV equal to the number of frames in the utterance, the value of LV is equal to that of GV.



(a) An example of GV calculation



(b) An example of LV calculation

Figure 4.2: Example of calculating GV and LV values for a scalar parameter sequence.

To more precisely model the dynamic characteristics of LV trajectories, we use a joint vector $\boldsymbol{z}$ of static and dynamic features of LVs as

$$\boldsymbol{z} = [\boldsymbol{z}_1^\top, ..., \boldsymbol{z}_t^\top, ..., \boldsymbol{z}_T^\top]^\top \tag{4.12}$$

$$\boldsymbol{z}_t = [\boldsymbol{v}_t^\top, \Delta^{(1)}\boldsymbol{v}_t^\top, \Delta^{(2)}\boldsymbol{v}_t^\top]^\top \tag{4.13}$$

where the dynamic features are defined in the same manner as Eq. (2.11), that is,

$$\Delta^{(n)}\boldsymbol{v}_t = \sum_{\tau=-L_-^{(n)}}^{L_+^{(n)}} w^{(n)}(\tau)\boldsymbol{v}_{t+\tau}, \qquad n = 1, 2. \tag{4.14}$$

The relationship between a joint feature vector $\boldsymbol{z}$ and a static feature vector $\boldsymbol{v}$ is represented by a linear transformation:

$$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{v} \tag{4.15}$$

where $\boldsymbol{W}$ is given by Eqs. (2.13)–(2.15).

4.3.2 Modeling of LV trajectories using HMMs

To use the LV as a variance compensation in parameter generation, we model LV trajectories by context-dependent phone hidden semi-Markov models (HSMMs) [18] that have explicit duration pdfs using the same framework as for the spectral parameter modeling. Specifically, first we calculate the LV feature vector $\boldsymbol{z}$ of spectral features for each utterance of the training data. Monophone HSMMs are then trained using the obtained LV feature vectors and corresponding phone labels with boundary information. Then, context-dependent HSMMs are trained from these monophone HSMMs using full-context labels including phonetic and prosodic contexts. We use the same monophone and full-context labels as those for modeling spectral parameters. The model parameters are tied using decision-tree-based context clustering. Finally, the model parameters of the tied HSMMs are re-estimated using the EM algorithm.

From the viewpoint of the synchrony between the state boundaries of spectral features and LV features, it would be better to simultaneously model spectral and LV models. A straightforward way to achieve the synchrony is to use the joint vector of these features for the model training. However, the number of dimensions of the feature vector becomes twice. This makes the computational time much longer and makes the feature space sparser, which is not always a desirable effect. Moreover, we confirmed that using the joint vector made the mel-cepstral distance between the original and generated parameters slightly larger than modeling each vector separately on the basis of a preliminary experimental result. Consequently the models of spectral parameters and their LVs are trained separately to reduce the computational costs in this study.

4.3.3 Parameter generation algorithm using LV model

The context-dependent LV models obtained in Section 4.3.2 are taken into account in the parameter generation process. The output pdf of the LV model, given an HSMM state sequence $\boldsymbol{q}_{\boldsymbol{z}}$, is given by a single multivariate

Gaussian pdf, that is,

$$P(\boldsymbol{z}|\boldsymbol{q}_{\boldsymbol{z}}, \boldsymbol{\lambda}_{\mathrm{LV}}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}_{\boldsymbol{q}_{\boldsymbol{z}}}, \boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}) \tag{4.16}$$

$$= \frac{1}{\sqrt{(2\pi)^{3DT}|\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}|}} \exp\left\{-\frac{1}{2}(\boldsymbol{z}-\boldsymbol{\mu}_{\boldsymbol{q}_{\boldsymbol{z}}})^{\top}\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_{\boldsymbol{q}_{\boldsymbol{z}}})\right\} \tag{4.17}$$

where $\boldsymbol{\lambda}_{\mathrm{LV}}$ denotes an LV model parameter set consisting of a mean vector $\boldsymbol{\mu}_{\boldsymbol{q}_{\boldsymbol{z}}}$ and a covariance matrix $\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}$. Since duration models of spectral and LV parameters are trained separately and the total numbers of frames of $\boldsymbol{q}$ and $\boldsymbol{q}_{\boldsymbol{z}}$ are not always equal, we use $\boldsymbol{q}$ for both state sequences of spectral and LV parameters in this study.

In the synthesis process, the optimum spectral parameter sequence $\hat{\boldsymbol{c}}$ is obtained so as to maximize the product of the pdfs for the spectral parameters and their LVs as follows:

$$\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{c}} P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda})^{\omega_{\mathrm{LV}}} P(\boldsymbol{z}|\boldsymbol{q}_{\boldsymbol{z}}, \boldsymbol{\lambda}_{\mathrm{LV}}) \tag{4.18}$$

where the constant ω_{LV} denotes a weight for controlling the balance between the two pdfs. In Eq. (4.18), the pdf $P(\boldsymbol{z}|\boldsymbol{q}_{\boldsymbol{z}}, \boldsymbol{\lambda}_{\mathrm{LV}})$ can be viewed as a penalty term for the reduction of the LV of the generated parameter sequence. The objective function $\mathcal{L}_{\boldsymbol{q}}^{(\mathrm{LV})}$ to be maximized with respect to $\boldsymbol{c}$ is written as

$$\mathcal{L}_{\boldsymbol{q}}^{(\mathrm{LV})} = \log P(\boldsymbol{o}|\boldsymbol{q}, \boldsymbol{\lambda})^{\omega_{\mathrm{LV}}} + \log P(\boldsymbol{z}|\boldsymbol{q}_{\boldsymbol{z}}, \boldsymbol{\lambda}_{\mathrm{LV}}) \tag{4.19}$$

$$\propto \omega_{\mathrm{LV}} \left(-\frac{1}{2}\boldsymbol{o}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{o} + \boldsymbol{o}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}}\right) - \frac{1}{2}\boldsymbol{z}^{\top}\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}^{-1}\boldsymbol{z} + \boldsymbol{z}^{\top}\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}_{\boldsymbol{z}}} \tag{4.20}$$

$$= \omega_{\mathrm{LV}} \left(-\frac{1}{2}\boldsymbol{c}^{\top}\boldsymbol{W}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{W}\boldsymbol{c} + \boldsymbol{c}^{\top}\boldsymbol{W}^{\top}\boldsymbol{U}_{\boldsymbol{q}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}}\right) - \frac{1}{2}\boldsymbol{v}^{\top}\boldsymbol{W}^{\top}\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}^{-1}\boldsymbol{W}\boldsymbol{v} + \boldsymbol{v}^{\top}\boldsymbol{W}^{\top}\boldsymbol{U}_{\boldsymbol{q}_{\boldsymbol{z}}}^{-1}\boldsymbol{\mu}_{\boldsymbol{q}_{\boldsymbol{z}}} \tag{4.21}$$

$$= \omega_{\mathrm{LV}} \left(-\frac{1}{2}\boldsymbol{c}^{\top}\boldsymbol{R}_{\boldsymbol{q}}\boldsymbol{c} + \boldsymbol{c}^{\top}\boldsymbol{r}_{\boldsymbol{q}}\right) - \frac{1}{2}\boldsymbol{v}^{\top}\boldsymbol{R}_{\boldsymbol{q}_{\boldsymbol{z}}}\boldsymbol{v} + \boldsymbol{v}^{\top}\boldsymbol{r}_{\boldsymbol{q}_{\boldsymbol{z}}} \tag{4.22}$$

where

$$\boldsymbol{R_q} = \boldsymbol{W}^\top \boldsymbol{U_q}^{-1} \boldsymbol{W} \tag{4.23}$$

$$\boldsymbol{r_q} = \boldsymbol{W}^\top \boldsymbol{U_q}^{-1} \boldsymbol{\mu_q} \tag{4.24}$$

$$\boldsymbol{R_{q_z}} = \boldsymbol{W}^\top \boldsymbol{U_{q_z}}^{-1} \boldsymbol{W} \tag{4.25}$$

$$\boldsymbol{r_{q_z}} = \boldsymbol{W}^\top \boldsymbol{U_{q_z}}^{-1} \boldsymbol{\mu_{q_z}}. \tag{4.26}$$

Since there is no closed-form solution for maximizing the objective function, an iterative optimization process for updating $\hat{\boldsymbol{c}}$ is necessary. We employ a gradient method as follows:

$$\hat{\boldsymbol{c}}^{(i+1)} = \hat{\boldsymbol{c}}^{(i)} + \alpha \cdot \delta \hat{\boldsymbol{c}}^{(i)} \tag{4.27}$$

where α is a step size parameter. We use a steepest descent method to calculate the vector $\delta \hat{\boldsymbol{c}}^{(i)}$. In the steepest descent algorithm, $\delta \hat{\boldsymbol{c}}^{(i)}$ is written as

$$\delta \hat{\boldsymbol{c}}^{(i)} = \left. \frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(\mathrm{LV})}}{\partial \boldsymbol{c}} \right|_{\boldsymbol{c}=\hat{\boldsymbol{c}}^{(i)}}. \tag{4.28}$$

The first derivative is calculated by

$$\frac{\partial \mathcal{L}_{\boldsymbol{q}}^{(\mathrm{LV})}}{\partial \boldsymbol{c}} = (-\boldsymbol{R_q c} + \boldsymbol{r_q}) + \omega_{\mathrm{LV}} \boldsymbol{x} \tag{4.29}$$

$$\boldsymbol{x} = [\boldsymbol{x}_1^\top, \boldsymbol{x}_2^\top, ..., \boldsymbol{x}_t^\top, ..., \boldsymbol{x}_T^\top]^\top \tag{4.30}$$

$$\boldsymbol{x}_t = [x_t(1), ..., x_t(d), ..., x_t(D)]^\top \tag{4.31}$$

$$x_t(d) = -\frac{2}{L} \left(c_t(d) - \overline{c(d)}_t \right) \left({\boldsymbol{v}_t}^\top \boldsymbol{R}_{q_{\boldsymbol{z}}(t)} - \boldsymbol{r}_{q_{\boldsymbol{z}}(t)} \right) \tag{4.32}$$

where $\boldsymbol{R}_{q_{\boldsymbol{z}}(t)}$ is the t-th column vector of the matrix $\boldsymbol{R}_{q_{\boldsymbol{z}}}$ and $\boldsymbol{r}_{q_{\boldsymbol{z}}(t)}$ is a component vector of $\boldsymbol{r}_{q_{\boldsymbol{z}}}$ at frame t.

To give the initial value of the static feature vector sequence $\hat{\boldsymbol{c}}^{(0)}$, we use the same manner as in the conventional parameter generation algorithm with GV [14]. Specifically, the spectral parameter sequence $\boldsymbol{c}'$ which is linearly converted from the maximum likelihood estimate is used as follows:

$$c'_t(d) = \sqrt{\frac{\mu_{\mathrm{g}}(d)}{v(d)}} \left(c_t(d) - \overline{c(d)} \right) + \overline{c(d)} \tag{4.33}$$

where ${c'}_t(d)$ is the d-th element of $\boldsymbol{c}'$ at frame t, and $\mu_g(d)$ is the d-th element of the mean vector $\boldsymbol{\mu}_g$ of the GV model.

4.4 Experiments

4.4.1 Experimental conditions

We used speech data of four professional narrators, two males (MHT, MYI) and two females (FTK, FYM), taken from the ATR Japanese speech database set B. Each speaker uttered 503 phonetically balanced sentences [68]. The average duration of all utterances was 5.0 sec. In each speaker's data set, 450 sentences (subsets A to I) and 53 sentences (subset J) were used for training/tuning and testing, respectively. Speech signals were sampled at a rate of 16kHz, and STRAIGHT analysis [56] was used to extract spectral envelope, F0, and aperiodicity features with a 5 ms frame shift. The spectral envelope was converted to mel-cepstral coefficients using a recursive formula. The aperiodicity feature was converted to average values for five frequency sub-bands, i.e., 0-1, 1-2, 2-4, 4-6, and 6-8 kHz. The resultant feature vector consisted of 40 mel-cepstral coefficients including the zeroth coefficient, log F0, five average band aperiodicities, and their delta and delta-delta coefficients. The total number of dimension was 138. In calculating the LV, L_+ and L_- were set to $L_+ = L/2$ and $L_- = L/2-1$, respectively. It is noted that only the mel-cepstral coefficients were used for the modeling of LVs, and the LV model was applied only to the spectral feature in parameter generation. The total number of dimension of the LV feature vector was 120.

A five-state left-to-right model structure with no skip was used for the HSMM. ω_{LV} of Eq. (4.18) was set to unity because the number of dimensions of vectors $\boldsymbol{o}$ and $\boldsymbol{z}$ are equal. In the decision-tree-based context clustering, the minimum description length (MDL) was used as a stopping criterion [23]. In the parameter generation, the spectral parameters of all dimensions were generated separately. The threshold for Eq. (4.28) was set to 0.0001 for both GV and LV cases. Maximum numbers of iteration for the gradient method were set to 50 for GV and 500 for LV.

4.4.2 Choice of window size for LV modeling

Before comparing the proposed technique to the conventional ones, we experimentally explored the choice of an appropriate window size for the proposed

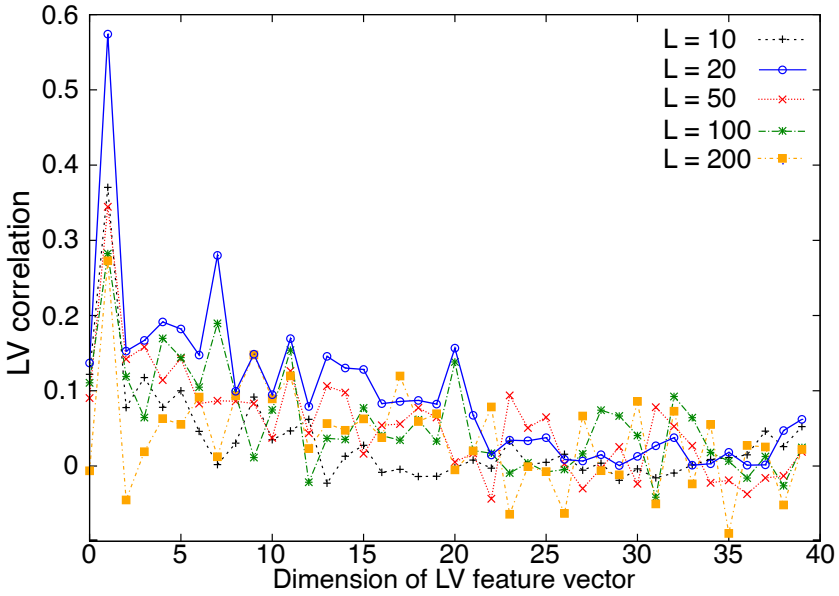


Figure 4.3: Average correlations between the original and generated LV trajectories with different window sizes for speaker MHT.

technique through objective and subjective evaluations using a development set. This is because the window size L for calculating the LVs should be determined in advance. In the experiments, we evaluated the speech data of two speakers: a male (MHT) and a female (FTK). In each speaker's data set, 400 sentences (subsets A to H) were used as a training set, and 50 sentences (subset I) were used as a development set to tune the appropriate window size. We generated mel-cepstral trajectories using the technique presented in Section 4.3.3 and calculated the correlations of the LVs between the original and generated mel-cepstral trajectories with different window sizes of 10, 20, 50, 100, and 200 frames. In calculating the correlations, we excluded silence frames in accordance with the label information.

Figs. 4.3 and 4.4 show the average correlations over all sentences for speakers MHT and FTK, respectively. The results for both speakers indicate that a window size of 20 frames gives the best correlations in the lower dimensions (less than 20) of the LV feature. To clarify the effect of the choice of window size on LV modeling, LV trajectories were calculated with different window sizes for the 1st coefficient of the original and generated mel-cepstral sequences using the conventional technique without GV modeling (noGV)

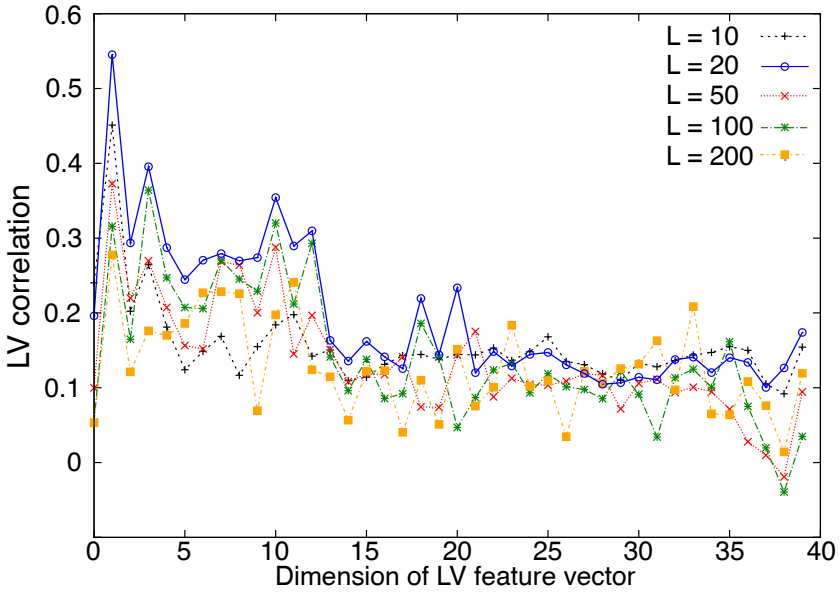


Figure 4.4: Average correlations between the original and generated LV trajectories with different window sizes for speaker FTK.

and the proposed technique (LV). Fig. 4.5 shows examples of these trajectories. In the figure, a window size of 20 frames gives a more appropriate LV trajectory than the others.

Next, we conducted a subjective evaluation test to confirm that the synthetic speech obtained with the chosen window size based on the objective measure gives better perceptual quality. We used an XAB listening test to evaluate the perceptual quality of synthetic speech in terms of its similarity to the vocoded speech. The participants were six Japanese native speakers, and six sentences were randomly chosen from the 50 test sentences for each participant. After given a vocoded speech sample (X) as a reference, the participants listened to two synthetic speech samples (A and B) in random order and were asked whether A or B was closer to X. For the pairs of A and B, we used synthetic speech samples with all ten combinations of the window sizes of 10, 20, 50, 100, and 200. Fig. 4.6 shows the average preference scores. It is seen that the scores with $L = 20$ were significantly higher than those with the other window sizes for both speakers. Taking these results of both objective and subjective evaluation into consideration, we fixed the window size to 20 frames in the following experiments. Table 4.1 lists the

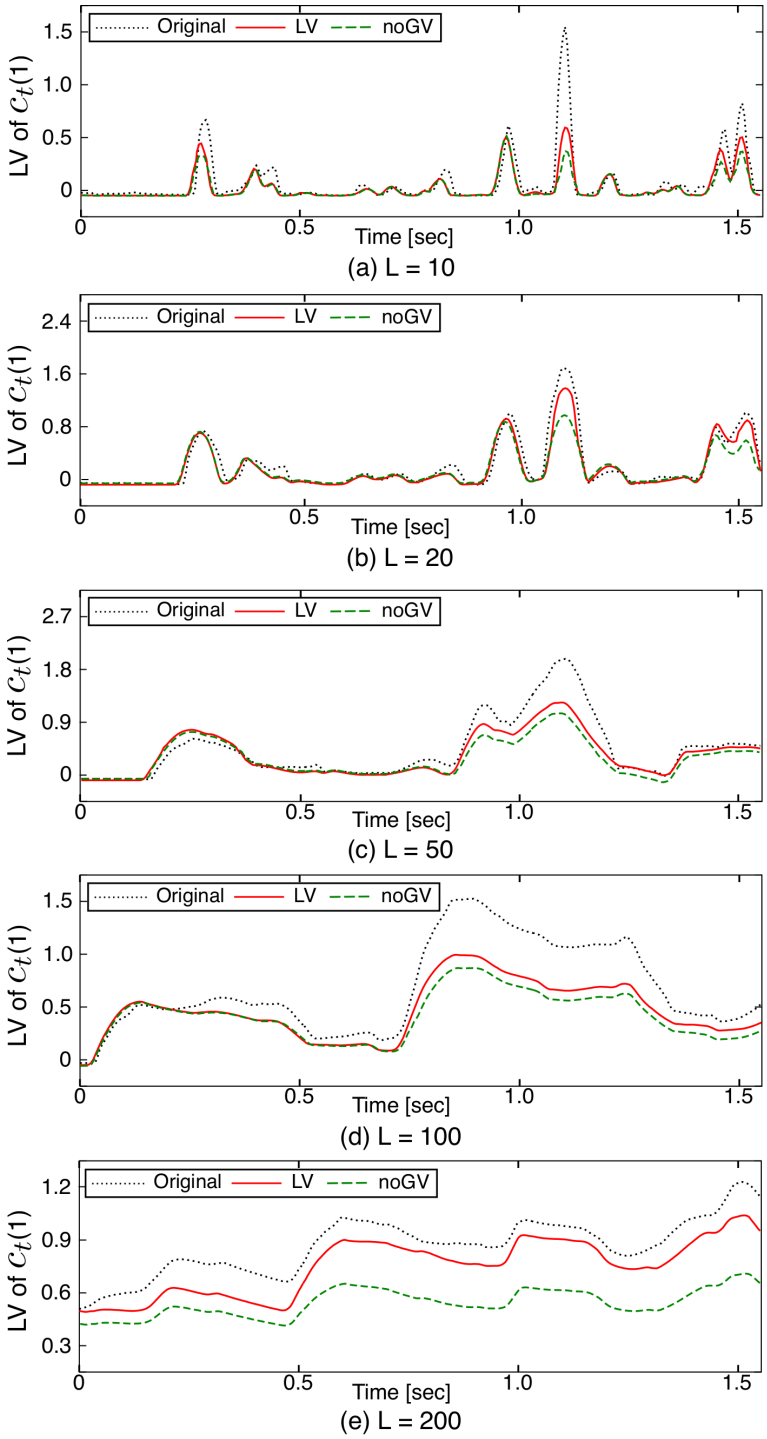


Figure 4.5: Examples of LV trajectories calculated from the original and generated sequences of the 1st mel-cepstral coefficient with different window sizes.

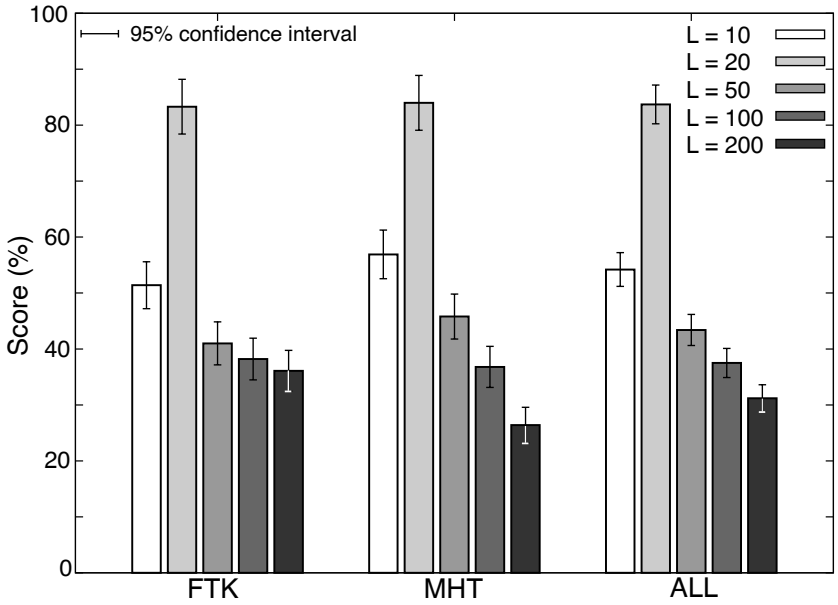


Figure 4.6: Results of XAB tests on synthetic speech reproducibility for all combinations of window sizes.

Table 4.1: Total number of leaf nodes in the decision tree.

Speaker	Mel-cepstrum	Log F0	Duration	LV (mcep)
MHT	766	1906	185	1537
MYI	510	1717	162	1005
FTK	587	1790	179	1243
FYM	592	1781	188	1136
Average	614	1799	179	1230

average values of the total numbers of leaf nodes of the decision trees for mel-cepstrum, log F0, duration, and LV when $L = 20$.

4.4.3 Comparison of parameter generation algorithms using GV and LV

For comparison, we evaluated synthetic speech samples obtained with three different parameter generation techniques in the synthesis stage: the tradi-

tional technique without a GV model (noGV), the conventional technique using a GV model (GV), and the proposed technique using the LV model (LV).

4.4.3.1 Objective evaluation results

First, we objectively evaluated the performance of the proposed technique by comparing it with the conventional techniques. For the objective distortion measures, we used the mel-cepstral distance and the RMS error of the LVs of the mel-cepstrum between the original and synthetic speech samples. In calculating distortion, phone boundary information of the original speech was used in parameter generation to align the mel-cepstral sequences. Silence frames were excluded in the calculation. We compared three types of synthetic speech obtained with different parameter generation algorithms, i.e., noGV, GV, and LV as described in Section 4.4.1. In this experiment, 450 sentences (subsets A to I) and 53 sentences (subset J) were used for training and testing, respectively.

Fig. 4.7 shows the average mel-cepstral distances for the respective speakers. From the figure, it is seen that the conventional GV-based parameter generation increased the spectral distortion for all speakers. This is because a single GV model is a very rough representation of variance characteristics, and dependence on linguistic contexts is not taken into account in the parameter generation process. By contrast, the proposed LV-based parameter generation gave significantly lower distortion than that of the GV-based parameter generation, and the distortion increase due to the variance compensation was suppressed. Fig. 4.8 shows the average RMS errors of LVs for respective speakers. It is seen that the GV-based parameter generation was not effective in compensating the LVs whereas the proposed technique gave the lowest RMS error among the three algorithms for all speakers. These results indicate that the proposed technique with context-dependent LV models can impose a more precise constraint of LV with a small increase in spectral distortion than the conventional techniques with and without a GV model.

Fig. 4.9 shows examples of LV trajectories calculated from the generated mel-cepstral sequences for the 1st and 3rd coefficients with each of three

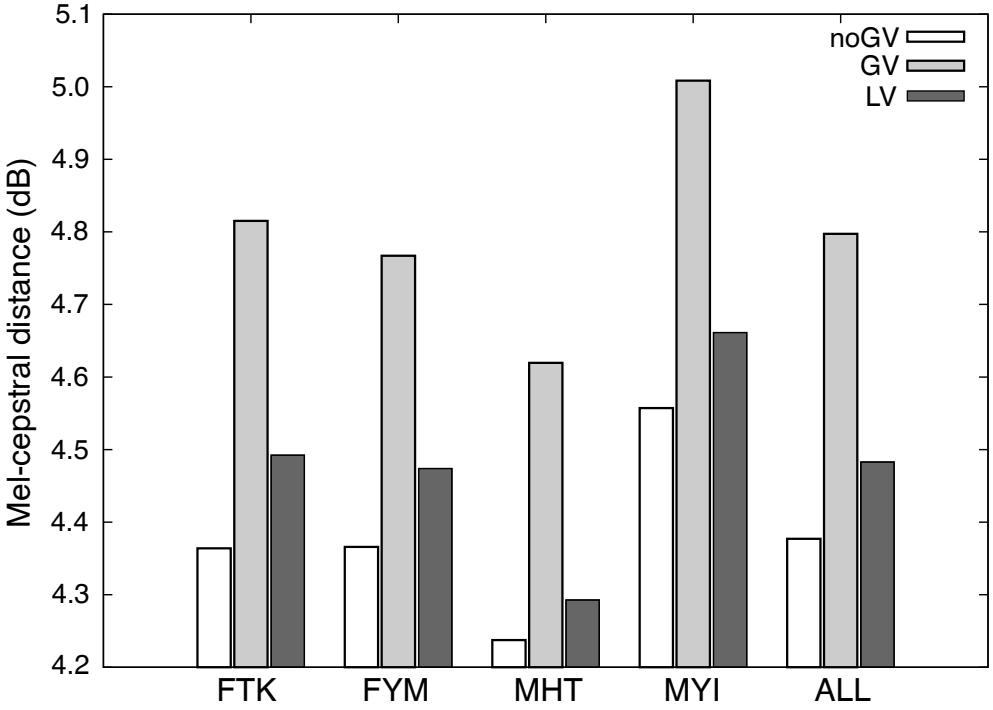


Figure 4.7: Average mel-cepstral distances between the original and synthetic speech.

algorithms. From the figure, it is seen that the LVs were also excessively enhanced in certain frames when GV-based compensation was used. On the other hand, the proposed technique did not cause such over-enhancement and made the LVs closer to those of the original mel-cepstral sequence than the other algorithms.

4.4.3.2 Subjective evaluation results

Finally, we conducted subjective evaluation tests for synthetic speech with the noGV, GV, and LV techniques described in Section 4.4.1. In this experiment, we only evaluated speech data from the male speaker MHT. 450 sentences (subsets A to I) and 53 sentences (subset J) were used for training and testing, respectively. Participants were seven Japanese native speakers, and eight sentences were randomly chosen from the 53 test sentences for each participant.

First, we evaluated the similarity of the synthetic speech to the original

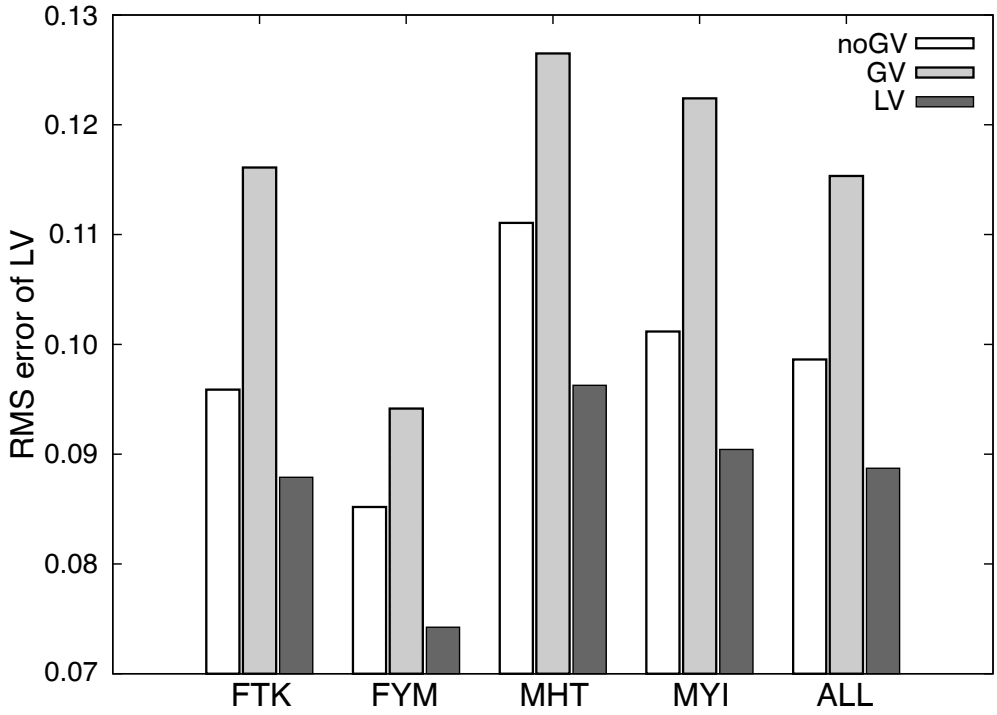


Figure 4.8: Average RMS errors of the LVs between the original and synthetic speech.

one with a degradation mean opinion score (DMOS) test. Vocoded speech was used as a reference. The participants listened to the reference and synthetic speech samples in that order and evaluated the perceived difference between the two samples on a 5-point scale: "1" for very annoying, "2" for annoying, "3" for slightly annoying, "4" for audible but not annoying, and "5" for inaudible. Fig. 4.10 shows the average scores for respective techniques with confidence intervals of 95%. From the results, GV was found to be not always effective in improving the similarity of the synthetic speech to the original one. One reason is that the GV-based parameter generation distorted the spectral parameters, as shown in Fig. 4.7, and this distortion could degrade the subjective similarity. In contrast, we see that the proposed LV-based parameter generation significantly improved the reproducibility of the original speech when comparing with the noGV and GV cases.

Next, we evaluated the naturalness of the synthetic speech samples with a mean opinion score (MOS) test. The participants listened to the synthetic

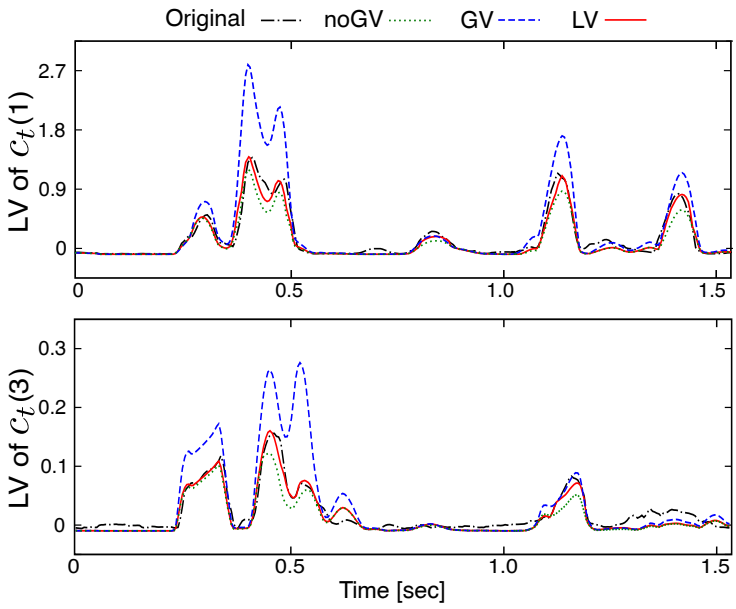


Figure 4.9: Examples of LV trajectories calculated from the original and generated sequences of the 1st and 3rd mel-cepstral coefficients.

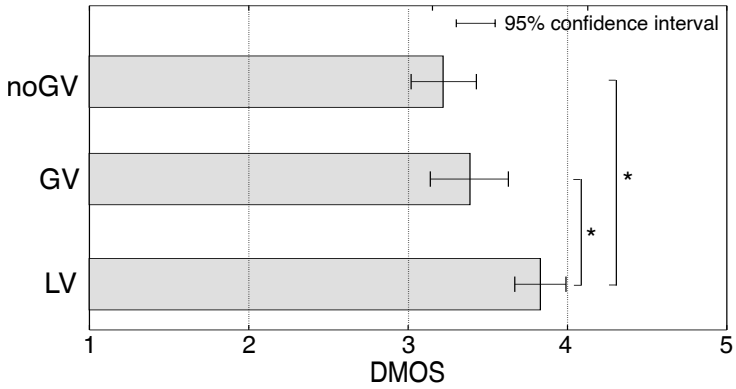


Figure 4.10: Results of a DMOS test on the similarity between the original and synthetic speech. Symbol * means that there is a statistically significant difference at a 1% significance level.

speech samples and evaluated their naturalness on a 5-point scale: "1" for bad, "2" for poor, "3" for fair, "4" for good, and "5" for excellent. Fig. 4.11 shows the average scores for respective techniques with confidence intervals of 95%. From the results, we see that introducing the LV model into the parameter generation significantly improved the naturalness of the synthetic

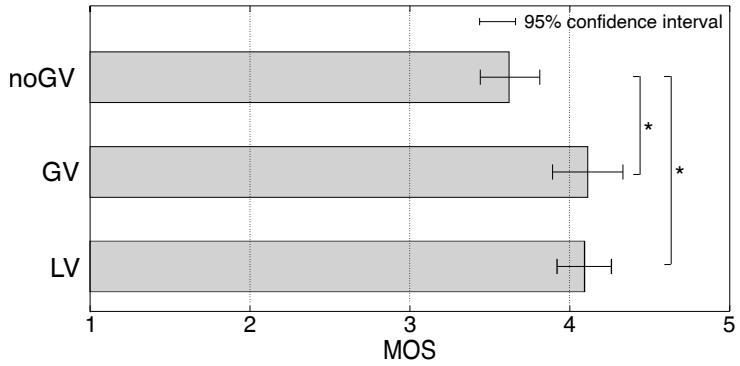


Figure 4.11: Results of a MOS test on the naturalness of the synthetic speech. Symbol * means that there is a statistically significant difference at a 1% significance level.

Table 4.2: Average computational time for one utterance with each method. The average number of frames in each utterance was 793.

	Average time (ms)	Ratio
noGV	74	1.0
GV	170	2.3
LV	287	3.9

speech as well as GV when comparing LV with noGV. Since there is no significant difference between the LV and GV results, the naturalness of the synthetic speech with LV and GV seems to be comparable.

From both results, it is concluded that the proposed technique significantly improves the reproducibility of original speech compared to the conventional techniques without degrading the naturalness of the synthetic speech.

4.4.3.3 Computational cost

4.4.3.3. To estimate the computational cost of the proposed parameter generation algorithm, we calculated the computation times of respective techniques for all the test sentences of speaker MHT. This experiment was performed on a machine with an Intel U2700 CPU at 1.30GHz (single thread) and 3 GB

of memory. The average numbers of gradient steps for all dimensions in GV- and LV-based parameter generation were 12 and 90, respectively. Table 4.2 shows the average computation times for generating a mel-cepstral sequence of an input sentence. From the results, it is seen that the proposed technique requires some additional computation time compared to the GV-based one. Specifically, the LV-based technique took 249 ms of processing time to synthesize an utterance with around 793 frames (about 4 sec). It is noted that the threshold and maximum number of iteration used in the gradient method for LV-based parameter generation was determined in an ad hoc manner and not tuned.

4.5 Conclusions

This chapter has proposed a parameter generation algorithm for spectral parameters using the LV model in HMM-based speech synthesis. The LV feature includes the dynamic features of LVs and is modeled using context-dependent HMMs. The performance was found to vary depending on the window size used for calculating LVs. Objective evaluation results showed that the proposed technique can significantly decrease the spectral distortion between the original and synthetic speech compared to conventional GV-based parameter generation. Subjective evaluation results showed that the subjective reproducibility was also significantly improved by using the proposed technique. Moreover, the naturalness of synthetic speech was improved by applying the proposed technique compared to the traditional parameter generation without the variance compensation as well as the GV-based parameter generation. In the future work, we will apply the proposed algorithm to F0 parameter generation. Although the increase of the computational cost of the proposed technique from the GV-based one is not so remarkable compared to the GV case, reducing the computation time by tuning the convergence criterion will be necessary for mobile applications. It will also not be a negligible issue to examine perceptually appropriate weights for both GV- and LV-based parameter generation and comparing their performance.

Chapter 5

Conclusions and Future Work

5.1 Summary of the thesis

Statistical parametric speech synthesis based on HMM is an effective framework for generating stable and diverse synthetic speech and has been widely studied. Specifically, this approach enables us not only to produce smooth and stable speech in a small footprint but also to add more variations to synthetic speech by using a variety of techniques: adaptation, interpolation, and control techniques for speaker characteristics, emotional expressions, speaking styles, and so on.

In order to improve the performance of HMM-based statistical parametric speech synthesis, we addressed two issues in speech synthesis in this thesis. One is the tone correctness in tonal language. The other is the spectral reproducibility in parameter generation process.

We first described the principle of the HMM-based statistic parametric speech synthesis in Chapter 2. The basic structure and algorithms of the HMM-based TTS system are also briefly described. The various techniques in each module were reviewed. We also described the drawbacks and refinements of statistical parametric synthesis approach.

In Chapter 3, we proposed a tone-modeling approach using the quantized F0 context in tonal language based on an average voice model trained with non-professional speech corpus in order to improve the tone correctness of synthetic speech. We attempted to reduce tone disagreements in speech

data acquired from nonprofessional speakers without manually modifying the labels. It is conducted by utilizing quantized F0 context as the tonal context in order to obtain an appropriate F0 model. We extracted the tonal context from real speech directly to prevent the tone disagreement between speech data and tone labels generated from transcription. Two methods of tone context labeling including the quantized F0 symbols based on phone and sub-phone boundaries have been investigated. The experimental results showed that our technique could bring about the improvement of tone correctness with a satisfactory level. Both objective and subjective tests indicated that the proposed technique can improve the naturalness and the tone correctness of the synthetic speech under condition of using an average voice model and a small amount of speech data of nonprofessional speakers.

In chapter 4, we investigated a technique for modeling LV of speech features and introduced LV into the parameter generation stage for HMM-based speech synthesis to reduce the over-smoothing problem of spectral reproducibility. The LV feature includes the dynamic features of LVs and is modeled using context-dependent HMMs. In the synthesis stage, a spectral parameter sequence is estimated so as to maximize a target function where conventional HMMs and LV models are combined. The experimental results showed that the proposed technique can decrease the spectral distortion between the synthetic and original speech compared to the conventional GV-based parameter generation. From the subjective evaluation results, we showed that the perceptual reproducibility was also significantly improved by using the proposed technique. Moreover, the quality of synthesized speech was improved when using our proposed method compared to the traditional parameter generation without the variance compensation, which is similar to the effectiveness of considering GV in parameter generation process.

5.2 Future work

In this study, we evaluated the proposed techniques for only the reading style speech data. Future work will focus on investigations of the synthetic speech with greater varieties of speech types such as spontaneous speech and expressive speech. We have to prepare the speech database with these kinds

of speaking styles and emotional expressions. Some successful study for this field is based on model interpolation and model adaptation techniques called style interpolation and style adaptation [69], [70].

Applying the proposed tonal-modeling approach with quantized F0 symbols to other tonal language is possible due to the simplicity and independence of language. In the tonal languages, the tone sandhi for standard Thai [71] is expected to be studied and incorporated in our system. We expect that it helps to treat the tone distortion problem in the further step. Moreover, other model adaptation techniques, such as SMAPLR (structural MAPLR) [72], CMLLR (Constrained MLLR) [73] [74], and CSMAPLR [75] adaptation, could be investigated and applied to accomplishing further improvements.

In the parameter generation using LV constraint approach, we applied only in the spectral parameter generation part. Therefore, evaluation of the proposed techniques by applying to the F0 parameter generation part will be important. We will also examine the effect of using only static features of LV in parameter generation. Moreover, applying our proposed to other languages, such as Thai, English and Mandarin, will also be evaluated the performance.

Appendix A

Mathematical Proofs

The derivation of the objective function in Eq. (4.20) and the first derivative in Eq. (4.29) shown in Chapter 4 is shown in the following sections.

A.1 Derivation of Eq. (4.20)

The objective function given by Eq. (A.1) is rewritten as follows:

$$\mathcal{L}_q^{(LV)} = \log P(\boldsymbol{o}|\boldsymbol{q},\boldsymbol{\lambda})^{\omega_l} + \log P(\boldsymbol{z}|\boldsymbol{q}_z,\boldsymbol{\lambda}_{LV}) \tag{A.1}$$

$$\begin{aligned}
&\propto \omega_l \left(\log \left[\frac{1}{\sqrt{(2\pi)^{3DT}|\boldsymbol{U}_q|}} \exp\left(-\frac{1}{2} \left(\boldsymbol{o}-\boldsymbol{\mu}_q\right)^\top \boldsymbol{U}_q^{-1} \left(\boldsymbol{o}-\boldsymbol{\mu}_q\right) \right) \right] \right) \\
&\quad + \log \left[\frac{1}{\sqrt{(2\pi)^{3DT}|\boldsymbol{U}_{q_z}|}} \exp\left(-\frac{1}{2} \left(\boldsymbol{z}-\boldsymbol{\mu}_{q_z}\right)^\top \boldsymbol{U}_{q_z}^{-1} \left(\boldsymbol{z}-\boldsymbol{\mu}_{q_z}\right) \right) \right] \\
&= \omega_l \left(-\frac{1}{2} \left(\log 2\pi + \log|\boldsymbol{U}_q| + \left(\left(\boldsymbol{o}-\boldsymbol{\mu}_q\right)^\top \boldsymbol{U}_q^{-1} \left(\boldsymbol{o}-\boldsymbol{\mu}_q\right) \right) \right) \right) \\
&\quad + \left(-\frac{1}{2} \left(\log 2\pi + \log|\boldsymbol{U}_{q_z}| + \left(\boldsymbol{z}-\boldsymbol{\mu}_{q_z}\right)^\top \boldsymbol{U}_{q_z}^{-1} \left(\boldsymbol{z}-\boldsymbol{\mu}_{q_z}\right) \right) \right) \\
&= \omega_l \left(-\frac{1}{2} \left(\boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{o} - \boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{\mu}_q - \boldsymbol{\mu}_q^\top \boldsymbol{U}_q^{-1} \boldsymbol{o} + \boldsymbol{\mu}_q^\top \boldsymbol{U}_q^{-1} \boldsymbol{\mu}_q \right) \right) \\
&\quad + \left(-\frac{1}{2} \left(\boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{z} - \boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{\mu}_{q_z} - \boldsymbol{\mu}_{q_z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{z} + \boldsymbol{\mu}_{q_z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{\mu}_{q_z} \right) \right) \\
&= \omega_l \left(-\frac{1}{2} \left(\boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{o} - \boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{\mu}_q - \left(\boldsymbol{\mu}_q^\top \boldsymbol{U}_q^{-1} \boldsymbol{o} \right)^\top \right) \right) \\
&\quad + \left(-\frac{1}{2} \left(\boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{z} - \boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{\mu}_{q_z} - \left(\boldsymbol{\mu}_{q_z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{z} \right)^\top \right) \right) \\
&= \omega_l \left(-\frac{1}{2} \left(\boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{o} - 2 \left(\boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{\mu}_q \right) \right) \right) + \left(-\frac{1}{2} \left(\boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{z} - 2 \boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{\mu}_{q_z} \right) \right) \\
&= \omega_l \left(-\frac{1}{2} \boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{o} + \boldsymbol{o}^\top \boldsymbol{U}_q^{-1} \boldsymbol{\mu}_q \right) + \left(-\frac{1}{2} \boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{z} + \boldsymbol{z}^\top \boldsymbol{U}_{q_z}^{-1} \boldsymbol{\mu}_{qz} \right).
\end{aligned} \tag{A.2}$$

A.2 Derivation of Eq. (4.29)

The first derivative of the objective function is expressed as

$$\frac{\partial \mathcal{L}_q^{(LV)}}{\partial \boldsymbol{c}} = \omega_l \left(\frac{\partial}{\partial \boldsymbol{c}} (-\frac{1}{2} \boldsymbol{c}^\top \boldsymbol{R}_q \boldsymbol{c} + \boldsymbol{c}^\top \boldsymbol{r}_q) \right) + \frac{\partial}{\partial \boldsymbol{c}} \left(-\frac{1}{2} \boldsymbol{v}_l^\top \boldsymbol{R}_{q_z} \boldsymbol{v}_l + \boldsymbol{v}_l^\top \boldsymbol{r}_{q_z} \right). \tag{A.3}$$

Eq. (A.3) is rewritten as

$$\frac{\partial \mathcal{L}_q^{(LV)}}{\partial \boldsymbol{c}} = \omega_l \frac{\partial \boldsymbol{X}_1}{\partial \boldsymbol{c}} + \frac{\partial \boldsymbol{X}_2}{\partial \boldsymbol{c}} \tag{A.4}$$

where

$$\begin{aligned} \frac{\partial \boldsymbol{X}_1}{\partial \boldsymbol{c}} &= \frac{\partial}{\partial \boldsymbol{c}} \left(-\frac{1}{2} \boldsymbol{c}^\top \boldsymbol{R}_q \boldsymbol{c} + \boldsymbol{c}^\top \boldsymbol{r}_q \right) \\ &= \frac{\partial}{\partial \boldsymbol{c}} (-\frac{1}{2} \boldsymbol{c}^\top \boldsymbol{R}_q \boldsymbol{c}) + \frac{\partial}{\partial \boldsymbol{c}} (\boldsymbol{c}^\top \boldsymbol{r}_q) \\ &= -\frac{1}{2} (2 \boldsymbol{R}_q \boldsymbol{c}) + \boldsymbol{r}_q \\ &= -\boldsymbol{R}_q \boldsymbol{c} + \boldsymbol{r}_q \end{aligned} \tag{A.5}$$

$$\begin{aligned} \frac{\partial \boldsymbol{X}_2}{\partial \boldsymbol{c}} &= \frac{\partial}{\partial \boldsymbol{c}} \left(-\frac{1}{2} \boldsymbol{v}_l^\top \boldsymbol{R}_{q_z} \boldsymbol{v}_l + \boldsymbol{v}_l^\top \boldsymbol{r}_{q_z} \right) \\ &= \frac{\partial}{\partial \boldsymbol{c}} (-\frac{1}{2} \boldsymbol{v}_l^\top \boldsymbol{R}_{q_z} \boldsymbol{v}_l) + \frac{\partial}{\partial \boldsymbol{c}} (\boldsymbol{v}_l^\top \boldsymbol{r}_{q_z}) \\ &= \frac{\partial \boldsymbol{Y}_1}{\partial \boldsymbol{c}} + \frac{\partial \boldsymbol{Y}_2}{\partial \boldsymbol{c}}. \end{aligned} \tag{A.6}$$

The derivative $\frac{\partial \boldsymbol{Y}_1}{\partial \boldsymbol{c}}$ is calculated as

$$\frac{\partial \boldsymbol{Y}_1}{\partial \boldsymbol{c}} = \frac{\partial}{\partial \boldsymbol{c}} \left(-\frac{1}{2} \boldsymbol{v}_l^\top \boldsymbol{R}_{q_z} \boldsymbol{v}_l \right) \tag{A.7}$$

$$\begin{aligned} \boldsymbol{Y}_1 &= -\frac{1}{2} \boldsymbol{v}_l^\top \boldsymbol{R}_{q_z} \boldsymbol{v}_l \\ &= -\frac{1}{2} \sum_{i=1}^{D} \sum_{j=1}^{D} v_l(i) v_l(j) \boldsymbol{R}_{q_z}(i,j). \end{aligned} \tag{A.8}$$

From the chain rule, we obtain

$$\frac{\partial Y_1}{\partial c_t(d)} = \frac{\partial Y_1}{\partial v_t(d)} \cdot \frac{\partial v_t(d)}{\partial c_t(d)} \tag{A.9}$$

$$\begin{aligned}\frac{\partial Y_1}{\partial v_t(d)} &= -\frac{1}{2}\left(2\sum_{i=1}^{D} v_l(i)\boldsymbol{R}_{q_z}(i,d)\right) \\ &= -\boldsymbol{v}_{l(t)}^{\top}\boldsymbol{R}_{q_z(t)}\end{aligned} \tag{A.10}$$

$$\begin{aligned}\frac{\partial v_t(d)}{\partial c_t(d)} &= \frac{\partial}{\partial c_t(d)}\left(\frac{1}{L}(\sum_{i=t-L_-}^{t+L_+}(c_i(d)-\overline{c(d)}_t)^2)\right) \\ &= \frac{1}{L}\frac{\partial}{\partial c_t(d)}\left(c_t^2(d) - 2\overline{c(d)}_t c_t(d) + (\overline{c(d)}_t)^2\right) \\ &= \frac{1}{L}\left(2c_t(d) - 2\overline{c(d)}_t\right).\end{aligned} \tag{A.11}$$

Consequently, $\dfrac{\partial Y_1}{\partial c_t(d)}$ is written as

$$\frac{\partial Y_1}{\partial c_t(d)} = -\frac{2}{L}\left(c_i(d) - \overline{c(d)}_t\right)\boldsymbol{v}_{l(t)}^{\top}\boldsymbol{R}_{q_z(t)}. \tag{A.12}$$

Similarly, $\dfrac{\partial Y_2}{\partial c}$ is calculated by

$$\frac{\partial \boldsymbol{Y}_2}{\partial \boldsymbol{c}} = \frac{\partial}{\partial \boldsymbol{c}}(\boldsymbol{v}_l^{\top}\boldsymbol{r}_{q_z}) \tag{A.13}$$

$$\begin{aligned}\boldsymbol{Y}_2 &= \boldsymbol{v}_l^{\top}\boldsymbol{r}_{q_z} \\ &= \sum_{i=1}^{D}\sum_{j=1}^{D} v_l(i)\boldsymbol{r}_{q_z}(i,j)\end{aligned} \tag{A.14}$$

and from the chain rule,

$$\frac{\partial Y_2}{\partial c_t(d)} = \frac{\partial Y_2}{\partial v_t(d)} \cdot \frac{\partial v_t(d)}{\partial c_t(d)} \tag{A.15}$$

$$\begin{aligned}\frac{\partial Y_2}{\partial v_t(d)} &= \sum_{j=1}^{D}\boldsymbol{r}_{q_z}(d,j) \\ &= \boldsymbol{r}_{q_z(t)}.\end{aligned} \tag{A.16}$$

Thus $\frac{\partial Y_2}{\partial c_t(d)}$ is written as

$$\frac{\partial Y_2}{\partial c_t(d)} = \frac{2}{L}\left(c_t(d) - \overline{c(d)}_t\right) \boldsymbol{r}_{q_z(t)}. \tag{A.17}$$

As a consequence, $\frac{\partial X_2}{\partial c}$ is written as

$$\begin{aligned}\frac{\partial \boldsymbol{X}_2}{\partial \boldsymbol{c}} &= \frac{\partial \boldsymbol{Y}_1}{\partial \boldsymbol{c}} + \frac{\partial \boldsymbol{Y}_2}{\partial \boldsymbol{c}} \\ &= -\frac{2}{L}\left(c_t(d) - \overline{c(d)}_t\right) \boldsymbol{v}_{l(t)}^{\top} \boldsymbol{R}_{q_z(t)} + \frac{2}{L}\left(c_t(d) - \overline{c(d)}_t\right) \boldsymbol{r}_{q_z(t)} \\ &= -\frac{2}{L}\left(c_t(d) - \overline{c(d)}_t\right) (\boldsymbol{v}_{l(t)}^{\top} \boldsymbol{R}_{q_z(t)} - \boldsymbol{r}_{q_z(t)}).\end{aligned} \tag{A.18}$$

Finally, $\frac{\partial \mathcal{L}_q^{(LV)}}{\partial \boldsymbol{c}}$ is written as

$$\begin{aligned}\frac{\partial \mathcal{L}_q^{(LV)}}{\partial \boldsymbol{c}} &= \omega_l \frac{\partial \boldsymbol{X}_1}{\partial \boldsymbol{c}} + \frac{\partial \boldsymbol{X}_2}{\partial \boldsymbol{c}} \\ &= \omega_l \left(-\boldsymbol{R}_q \boldsymbol{c} + \boldsymbol{r}_q\right) + [x_t(1), ..., x_t(d), ..., x_t(D)]^{\top}]\end{aligned} \tag{A.19}$$

where

$$x_t(d) = -\frac{2}{L}\left(c_t(d) - \overline{c(d)}_t\right) \left(\boldsymbol{v}_{l(t)}{}^{\top} \boldsymbol{R}_{q_z(t)} - \boldsymbol{r}_{q_z(t)}\right). \tag{A.20}$$

Bibliography

[1] J. Yamagishi, "Average-voice-based speech synthesis," Ph.D. Dissertation, Tokyo Institute of Technology, March 2006.

[2] H. Mixdorff, "Intonation patterns of German - Model-based quantitative analysis and synthesis of F0-contours," Ph.D. Dissertation, TU Dresden, 1998.

[3] A. Thangthai, N. Thatphithakkul, C. Wutiwiwatchai, A. Rugchatjaroen, and S. Saychum, "T-Tilt: a modified Tilt model for F0 analysis and synthesis in tonal languages," Proc. INTERSPEECH 2008, pp.2270–2273, Sept. 2008.

[4] A. Thangthai, A. Rugchatjaroen, N. Thatphithakkul, A. Chotimongkol, and C. Wutiwiwatchai, "Optimization of T-Tilt F0 modeling," Proc. INTERSPEECH 2009, pp.508–511, 2009.

[5] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," Speech Commununication, vol.9, no.5-6, pp.453–467, Dec. 1990.

[6] A.W. Black and N. Campbell, "Optimising selection of units from speech databases for concatenative synthesis," Proc. EUROSPEECH 1995, pp.581–584, Sept. 1995.

[7] 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 1999, pp.2347–2350, Sept. 1999.

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

[9] M.N. H. Zen, T. Toda and T. Tokuda, "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, 2007.

[10] 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 1995, pp.757–760, Sept. 1995.

[11] C. Wutiwiwatchai and S. Furui, "Thai speech processing technology: A review," Speech Communication, vol.49, no.1, pp.8–27, 2007.

[12] P. Mittrapiyanuruk, C. Hansakunbuntheung, V. Tesprasit, and V. Sornlertlamvanich, "Issues in Thai text-to-speech synthesis: The NECTEC approach," NECTEC Annual Conference, pp.483–495, 2000.

[13] T. Toda and K. Tokuda, "Speech parameter generation algorithm considering global variance for HMM-based speech synthesis," Proc. INTERSPEECH 2005, pp.2801–2804, 2005.

[14] T. Toda and K. Tokuda, "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, May 2007.

[15] A.J. Hunt and A.W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," Proc. ICASSP 1996, pp.373–376, May 1996.

[16] A.W. Black and P. Taylor, "Automatically clustering similar units for unit selection in speech synthesis.," Proc. EUROSPEECH 1997, pp.601–604, 1997.

[17] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Multi-space probability distribution HMM," IEICE Trans. Inf. & Syst., vol.E85-D, no.3, pp.455–464, March 2002.

[18] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," IEICE Trans. Inf. & Syst., vol.E90-D, no.5, pp.825–834, May 2007.

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

[20] S.E. Levinson, "Continuously variable duration hidden Markov models for automatic speech recognition," Computer Speech and Language, vol.1, no.1, pp.29–45, 1986.

[21] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Hidden semi-Markov model based speech synthesis," Proc. INTERSPEECH 2004-ICSLP, pp.1393–1396, Oct. 2004.

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

[23] K. Shinoda and T. Watanabe, "MDL-based context-dependent subword modeling for speech recognition," J. Acoust. Soc. Jpn. (E), vol.21, no.2, pp.79–86, March 2000.

[24] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Duration modeling for HMM-based speech synthesis," Proc. ICSLP-98, pp.29–32, Dec. 1998.

[25] K. Tokuda, T. Kobayashi, and S. Imai, "Speech parameter generation from HMM using dynamic features," Proc. ICASSP-95, pp.660–663, May 1995.

[26] S. Imai, K. Sumita, and C. Furuichi, "Mel log spectrum approximation (MLSA) filter for speech synthesis," IEICE Trans. A (Japanese Edition), vol.J66-A, no.2, pp.122–129, Feb. 1983.

[27] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," Proc. ICASSP-92, pp.137–140, March 1992.

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

[29] J. Yamagishi, T. Masuko, and T. Kobayashi, "MLLR adaptation for hidden semi-Markov model based speech synthesis," Proc. INTERSPEECH 2004-ICSLP, pp.1213–1216, Oct. 2004.

[30] J. Yamagishi, M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "A context clustering technique for average voice models," IEICE Trans. Inf. & Syst., vol.E86-D, no.3, pp.534–542, March 2003.

[31] J. Yamagishi, M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "A training method of average voice model for HMM-based speech synthesis," IEICE Trans. Fundamentals, vol.E86-A, no.8, pp.1956–1963, Aug. 2003.

[32] T. Dutoit, An Introduction to Text-to-Speech Synthesis, Kluwer Academic Publishers, 1997.

[33] H. Mixdorff, "A novel approach to the fully automatic extraction of Fujisaki model parameters," Proc. ICASSP 2000, pp.1281–1284, 2000.

[34] P. Seresangtakul and T. Takara, "A generative model of fundamental frequency contours for polysyllabic words of Thai tones," Proc. ICASSP 2003, pp.452–455, 2003.

[35] P.A. Taylor, "Analysis and synthesis of intonation using the Tilt model," Journal of the Acoustical Society of America, vol.107, no.3, pp.1697–1714, 2000.

[36] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Mixed excitation for HMM-based speech synthesis," Proc. EUROSPEECH, pp.2263–2266, Sept. 2001.

[37] L. Zhen-hu, W. Yi-jian, W. Yu-ping, Q. Long, and W. Ren-hua, "USTC system for blizzard challenge 2006 an improved HMM-based speech synthesis method," in Blizzard Challenge Workshop, 2006.

[38] Y. Kishimoto, H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Automatic estimation of postfilter coefficients for HMM-based speech synthesis," Proc. Spring Meeting of ASJ, pp.243–244, 2003.

[39] Y. Wu, H. Zen, Y. Nankaku, and K. Tokuda, "Minimum generation error criterion considering global/local variance for HMM-based speech synthesis," Proc. ICASSP 2008, pp.4621–4624, March 2008.

[40] J. Yamagishi, B. Usabaev, S. King, O. Watts, J. Dines, J. Tian, R. Hu, Y. Guan, K. Oura, K. Tokuda, R. Karhila, and M. Kurimo, "Thousands of voices for HMM-based speech synthesis: analysis and application of TTS systems built on various ASR corpora," IEEE Audio, Speech, & Language Processing, vol.18, no.5, pp.984–1004, July 2010.

[41] A. Raux and A.W. Black, "A unit selection approach to F0 modeling and its application to emphasis," Proc.Workshop on Automatic Speech Recognition and Understanding (ASRU), pp.700–705, 2003.

[42] Y. Li, T. Lee, and Y. Qian, "F0 analysis and modeling for Cantonese text-to-speech," Proc. Speech Prosody 2004, pp.169–180, March 2004.

[43] S. Chomphan and T. Kobayashi, "Tone correctness improvement in speaker dependent HMM-based Thai speech synthesis," Speech Communication, vol.50, no.5, pp.392–404, 2008.

[44] H. Fujisaki and K. Hirose, "Analysis of voice fundamental frequency contours for decorative sentence of Japanese," J. Acoust. Soc. Japan (ASJ), vol.5, no.4, pp.133–142, 1984.

[45] S. Chomphan and T. Kobayashi, "Tone correctness improvement in speaker-independent average-voice-based Thai speech synthesis," Speech Communication, vol.51, no.4, pp.330–343, 2009.

[46] P.A. Chou, "Optimal partitioning for classification and regression trees," IEEE Trans. Pattern Anal. Mach. Intell., vol.13, no.4, pp.340–354, April 1991.

[47] T. Nose, K. Ooki, and T. Kobayashi, "HMM-based speech synthesis with unsupervised labeling of accentual context based on F0 quantization and average voice model," Proc. ICASSP 2010, pp.4622–4625, March 2010.

[48] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR," Proc. ICASSP 2001, pp.805–808, May 2001.

[49] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "Text-to-speech synthesis with arbitrary speaker's voice from average voice," Proc. EUROSPEECH 2001, pp.345–348, Sept. 2001.

[50] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," IEICE Trans. Inf. & Syst., vol.E90-D, no.2, pp.533–543, Feb. 2007.

[51] K. Ogata, M. Tachibana, J. Yamagishi, and T. Kobayashi, "Acoustic model training based on linear transformation and MAP modification for HSMM-based speech synthesis," Proc. INTERSPEECH 2006-ICSLP, pp.1328–1331, Sept. 2006.

[52] S. Kasuriya, V. Sornlertlamvanich, P. Cotsomrong, S. Kanokphara, and N. Thatphithakkul, "Thai speech corpus for Thai speech recognition," Proc. Oriental COCOSDA 2003, pp.54–61, June 2003.

[53] C. Hansakunbuntheung, A. Rugchatjaroen, and C. Wutiwiwatchai, "Space reduction of speech corpus based on quality perception for unit selection speech synthesis," Proc. SNLP 2005, pp.127–132, Dec. 2005.

[54] V. Sornlertlamvanich, N. Takahashi, and H. Isahara, "Thai part-of-speech tagged corpus: ORCHID," Proc. Oriental COCOSDA 1998, pp.131–138, May 1998.

[55] H. Kawahara, "Speech representation and transformation using adaptive interpolation of weighted spectrum: Vocoder revisited," Proc. ICASSP 1997, pp.1303–1306, 1997.

[56] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," Speech Communication, vol.27, no.3-4, pp.187–207, 1999.

[57] C. Benoît, M. Grice, and V. Hazan, "The SUS test: A method for the assessment of text-to-speech synthesis intelligibility using semantically unpredictable sentences," Speech Communication, vol.18, no.4, pp.381–392, 1996.

[58] T. Nose and T. Kobayashi, "Recent development of HMM-based expressive speech synthesis and its applications," Proc. APSIPA ASC 2011, 2011. http://www.apsipa.org/proceedings_2011/pdf/APSIPA189.pdf.

[59] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Incorporating a mixed excitation model and postfilter into HMM-based text-to-speech synthesis," Systems and Computers in Japan, vol.36, no.12, pp.43–50, 2005.

[60] J. Yamagishi, H. Zen, Y. Wu, T. Toda, and K. Tokuda, "The HTS-2008 system: Yet another evaluation of the speaker-adaptive HMM-based speech synthesis system in the 2008 Blizzard Challenge," 2008.

[61] Z. Ling, Y. Hu, and L. Dai, "Global variance modeling on the log power spectrum of LSPs for HMM-based speech synthesis," Proc. INTERSPEECH 2010, pp.825–828, Sept. 2010.

[62] S. Pan, Y. Nankaku, K. Tokuda, and J. Tao, "Global variance modeling on frequency domain delta LSP for HMM-based speech synthesis," Proc. ICASSP 2011, pp.4716–4719, May 2011.

[63] H. Zen, M. Gales, Y. Nankaku, and K. Tokuda, "Product of experts for statistical parametric speech synthesis," IEEE Trans. Audio, Speech, and Language Process., vol.20, no.3, pp.794–805, 2012.

[64] G. Hinton, "Products of experts," Proc. ICANN 99, pp.1–6, 1999.

[65] H. Zen, K. Tokuda, and T. Kitamura, "Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences," Computer Speech & Language, vol.21, no.1, pp.153–173, 2007.

[66] Y. Wu and R. Wang, "Minimum generation error training for HMM-based speech synthesis," Proc. ICASSP 2006, pp.889–892, May 2006.

[67] V. Chunwijitra, T. Nose, and T. Kobayashi, "A speech parameter generation algorithm using local variance for HMM-based speech synthesis," Proc. INTERSPEECH 2012, pp.1151–1154, 2012.

[68] 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 Communication, vol.9, no.4, pp.357–363, 1990.

[69] M. Tachibana, J. Yamagishi, T. Masuko, and T. Kobayashi, "Speech synthesis with various emotional expressions and speaking styles by style interpolation and morphing," IEICE Trans. Inf. Syst., vol.E88-D, no.11, pp.2484–2491, Nov. 2005.

[70] M. Tachibana, J. Yamagishi, T. Masuko, and T. Kobayashi, "A style adaptation technique for speech synthesis using HSMM and suprasegmental features," IEICE Trans. Inf. Syst., vol.E89-D, no.3, pp.1092–1099, March 2006.

[71] N.G.I. Thompson, "Tone sandhi between complex tones in a seven-tone southern Thai dialect," Proc.ICSLP-98, pp.53–56, Dec. 1998.

[72] O. Siohan, T. Myrvoll, and C. Lee, "Structural maximum a posteriori linear regression for fast HMM adaptation," Computer Speech and Language, vol.16, no.3, pp.5–24, 2002.

[73] V. Digalakis, D. Rtischev, and L. Neumeyer, "Speaker adaptation using constrained estimation of gaussian mixtures," IEEE Trans. Speech Audio Processing, vol.3, no.5, pp.357–366, Sept. 1995.

[74] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," Computer Speech and Language, vol.12, pp.75–98, 1998.

[75] Y. Nakano, M. Tachibana, J. Yamagishi, and T. Kobayashi, "Constrained structural maximum a posteriori linear regression for average-voice-based speech synthesis," Proc. INTERSPEECH 2006, pp.2286–2289, Sept. 2006.

List of Publications

Publications Related to This Thesis

Journal

1. Vataya Chunwijitra, Takashi Nose, Takao Kobayashi,
   "A tone-modeling technique using a quantized F0 context to improve tone correctness in average-voice-based speech synthesis,"
   Speech Communication, vol.54(2), pp.245-255 (2012.02).

2. Takashi Nose, Vataya Chunwijitra, Takao Kobayashi,
   "A parameter generation algorithm using local variance for HMM-based speech synthesis,"
   submitted to IEEE Journal of Selected Topics in Signal Processing.

International Conference

1. Vataya Chunwijitra, Takashi Nose, Takao Kobayashi,
   "Tonal context labeling using quantized F0 symbols for improving tone correctness in average-voice-based speech synthesis,"
   Proc. 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2011, pp.4708-4711 (2011.05).

2. Vataya Chunwijitra, Takashi Nose, Takao Kobayashi,
   "A speech parameter generation algorithm using local variance for HMM-based speech synthesis,"
   Proc. 13th Annual Conference of the International Speech Communication Association, INTERSPEECH 2012, (2012.09).