DOCTORAL DISSERTATION

SIMULTANEOUS MODELING OF PHONETIC AND PROSODIC PARAMETERS, AND CHARACTERISTIC CONVERSION FOR HMM-BASED TEXT-TO-SPEECH SYSTEMS

DOCTOR OF ENGINEERING

JANUARY 2002

TAKAYOSHI YOSHIMURA

Supervisors : Professor Tadashi Kitamura
Associate Professor Keiichi Tokuda

Department of Electrical and Computer Engineering
Nagoya Institute of Technology

Acknowledgement

Firstly, I would like to express my sincere gratitude to Professor Tadashi Kitamura and Associate Professor Keiichi Tokuda, Nagoya Institute of Technology, my thesis advisor, for their support, encouragement, and guidance. Also, I would like to express my gratitude to Professor Takao Kobayashi and Research Associate Takashi Masuko, Tokyo Institute of Technology, for their kind suggestion.

I would also like to thank all the members of Kitamura Laboratory and for their technical support, encouragement. If somebody was missed among them, my work would not be completed.

I would be remiss if I would fail to thank Rie Matae, the secretary to Kitamura Laboratory and Fumiko Fujii, the secretary to the Department of Computer Science, for their kind assistance.

Finally, I would sincerely like to thank my family for their encouragement.

Contents

Acknowledgement **i**

Contents **ii**

List of Tables **vi**

List of Figures **vii**

1 Introduction **4**

1.1 Text-to-speech(TTS) system 4

1.2 Proposition of this thesis 5

1.3 Original contributions 7

2 Mel-cepstral Analysis and Synthesis Technique **8**

2.1 Source-filter model 8

2.2 Mel-cepstral analysis 9

2.3 Synthesis filter 10

3 Speech Parameter Modeling Based on HMM **14**

3.1 Spectral parameter modeling 14

3.1.1 Continuous density HMM 14

3.1.2 Probability calculation 16

3.1.3 Parameter estimation of continuous density HMM 18

3.2 F0 parameter modeling 21

3.2.1 Multi-Space Probability Distribution 22

3.2.2 Multi-space distribution HMM 23
3.2.3 Reestimation algorithm for MSD-HMM training 25
3.2.4 Application to F0 pattern modeling 27

4 Speech parameter generation from HMM **29**
4.1 Speech parameter generation based on maximum likelihood criterion . 29
4.1.1 Case 1: Maximizing $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$ with respect to $\boldsymbol{O}$ 30
4.1.2 Case 2: Maximizing $P(\boldsymbol{O}, \boldsymbol{Q}|\lambda)$ with respect to $\boldsymbol{O}$ and $\boldsymbol{Q}$ . . . 31
4.1.3 Case 3: Maximizing $P(\boldsymbol{O}|\lambda)$ with respect to $\boldsymbol{O}$ 32
4.2 Example 33
4.2.1 Effect of dynamic feature 34
4.2.2 Parameter generation using multi-mixuture HMM 36

5 Construction of HMM-based Text-to-Speech System **39**
5.1 Calculation of dynamic feature 39
5.2 Spectrum and F0 modeling 40
5.3 Duration modeling 42
5.3.1 Overview 42
5.3.2 Training of state duration models 42
5.4 Context dependent model 43
5.4.1 Contextual factors 43
5.4.2 Decision-tree based context clustering 44
5.4.3 Context clustering using MDL principle 47

6 HMM-based Text-to-Speech Synthesis **49**
6.1 Overview 49
6.2 Text analysis 49
6.3 Duration determination 51
6.4 Speech parameter generation 51
6.5 Experiments 52
6.5.1 Effect of dynamic feature 52
6.5.2 Automatically system training 55

6.5.3 Speaking rate 56

7 Improvement of Synthesized Speech Quality 58

7.1 Introduction of Mixed Excitation Model 58

7.1.1 Mixed excitation 59

7.1.2 Excitation parameter modeling 62

7.1.3 Excitation parameter generation 64

7.2 Incorporation of postfilter 64

7.3 Experiments 65

7.3.1 Excitation generation 65

7.3.2 Effect of mixed excitation 65

7.3.3 Effect of postfiltering 67

8 Voice Conversion Technique: Speaker Interpolation 69

8.1 Overview 69

8.2 Speaker Interpolation 71

8.3 Simulation 73

8.4 Experiments 76

8.4.1 Generated Spectra 77

8.4.2 Experiment of Similarity 77

8.5 Discussion 78

9 Conclusions 80

9.1 Original contribution revisited 80

9.2 Future works 82

List of Publications 87

Journal Papers 87

International Conference Proceedings 87

Technical Reports 89

Domestic Conference Proceedings 89

Appendix A Examples of decision trees constructed by using the

MDL principle. **92**

List of Tables

2.1 Examples of α for approximating auditory frequency scales. 10

2.2 Coefficients of $R_4(\omega)$. 11

6.1 Number of distribution. 56

8.1 Setting for simulating Japanese vowel “a” 73

List of Figures

1.1 The scheme of the HMM-based TTS system. 6

2.1 Source-filter model. 9

2.2 Implementation of Synthesis filter $D(z)$. 13

3.1 Representation of a speech utterance using a five-state HMM. 15

3.2 Implementation of the computation using forward-backward algorithm in terms of a trellis of observation t and state i. 18

3.3 Example of F0 pattern. 21

3.4 Multi-space probability distribution and observations. 23

3.5 An HMM based on multi-space probability distribution. 24

4.1 Spectra generated with dynamic features for a Japanese phrase "chiisanaunagi". 34

4.2 Relation between probability density function and generated parameter for a Japanese phrase "unagi" (top: static, middle: delta, bottom: delta-delta). 35

4.3 Generated spectra for a sentence fragment "kiNzokuhiroo." 37

4.4 Spectra obtained from 1-mixture HMMs and 8-mixture HMMs. . . . 37

4.5 The result of the pair comparison test. 38

5.1 Calculation of dynamic features for F0. 40

5.2 Feature vector. 41

5.3 structure of HMM. 41

5.4 Decision trees. 45

5.5 Examples of decision trees. 46

6.1 The block diagram of the HMM-based TTS. 50

6.2 Generated spectra for a phrase “heikiNbairitsu” (top: natural spectra, bottom: generated spectra). 52

6.3 Generated F0 pattern for a sentence “heikiNbairitsuwo sageta keisekiga aru” (top: natural F0 pattern, bottom: generated F0 pattern). 53

6.4 Generated spectra for a phrase “heikiNbairitsu.” 53

6.5 Generated F0 pattern for a sentence “heikiNbairitsuwo sageta keisekiga aru.” 54

6.6 Effect of dynamic feature. 55

6.7 Effect of difference of initial model. 56

6.8 Generated spectra for an utterance “/t-o-k-a-i-d-e-w-a/” with different speaking rates (top : $\rho = -0.1$, middle : $\rho = 0$, bottom : $\rho = 0.1$). 57

7.1 Traditional excitation model. 59

7.2 Multi-band mixing model. 60

7.3 Mixed excitation model. 61

7.4 Structure of a feature vector modeled by HMM. 62

7.5 Structure of a HMM. 63

7.6 Effect of postfiltering(dots: before postfiltering, solid: after postfiltering $\beta = 0.5$). 65

7.7 Example of generated excitation for phrase “sukoshizutsu.” (top: traditional excitation, bottom: mixed excitation) 66

7.8 Comparison of traditional and mixed excitation models. 67

7.9 Effect of mixed excitation and postfiltering. 68

8.1 Block diagram of speech synthesis system with speaker interpolaiton. 70

8.2 A space of speaker individuality modeled by HMMs. 72

8.3 Comparison between method (a), (b) and (c) with regard to interpolation between two multi-dimensional Gaussian distributions. 74

8.4 Comparison between method (a), (b) and (c) with regard to interpolation between two Gaussian distributions p_1 and p_2 with interpolation ratios A: $(a_1, a_2) = (1, 0)$, B: $(a_1, a_2) = (0.75, 0.25)$, C: $(a_1, a_2) = (0.5, 0.5)$, D: $(a_1, a_2) = (0.25, 0.75)$, E: $(a_1, a_2) = (0, 1)$. . . 75

8.5 Generated spectra of the sentence "/n-i-m-o-ts-u/". 77

8.6 Subjective distance between samples. 78

A.1 Examples of decision trees for mel-cepstrum. 93

A.2 Examples of decision trees for F0. 94

A.3 Examples of decision trees for bandpass voicing strength. 95

A.4 Examples of decision trees for Fourier magnitude. 96

A.5 Examples of decision trees for duration. 97

Abstract

A text-to-speech(TTS) system is one of the human-machine interfaces using speech. In recent years, TTS system is developed as an output device of human-machine interfaces, and it is used in many application such as a car navigation system, information retrieval over the telephone, voice mail, a speech-to-speech translation system and so on. However, although most text-to-speech systems still cannot synthesize speech with various voice characteristics such as speaker individualities and emotions. To obtain various voice characteristics in text-to-speech systems based on the selection and concatenation of acoustical units, a large amount of speech data is necessary. However, it is difficult to collect, segment, and store it. From these points of view, in order to construct a speech synthesis system which can generate various voice characteristics, an HMM-based text-to-speech system has been proposed. This dissertation presents the construction of the HMM-based text-to-speech system, in which spectrum, fundamental frequency and duration are modeled simultaneously in a unified framework of HMM.

In the system, mainly three techniques are used; (1) a mel-cepstral analysis/synthesis technique, (2) speech parameter modeling using HMM and (3) a speech parameter generation algorithm from HMM. Since the system uses above three techniques, the system has several capabilities. First, since the TTS system uses the speech parameter generation algorithm, the generated spectral and pitch paramters from the trained HMMs can be similar to those of real speech. Second, by transforming HMM parameters appropriately, voice characteristics of synthetic speech can be changed since the system generates speech from the HMMs. Third, this system is trainable. In this thesis, first, the above three techniques are presented, and simultaneous modeling of phonetic and prosodic parameters in a framework of HMM is proposed.

Next, to improve of the quality of synthesized speech, the mixed excitation model of the speech coder MELP and postfilter are incorporated into the system. Experimental results show that the mixed excitation model and postfilter significantly improve the quality of synthesized speech.

Finally, for the purpose of synthesizing speech with various voice characteristics

such as speaker individualities and emotions, the TTS system based on speaker interpolation is presented.

Abstract in Japanese

テキスト音声合成 (TTS) システムは，音声を使ったマンマシンインターフェースの出力モジュールとして，近年，カーナビゲーションシステム，電話を利用した情報検索，音声翻訳システムなど様々なアプリケーションで用いられている．これまでに数多くのテキスト音声合成システムが提案され，音質的に向上している．しかし，まだ様々な話者の声質で話したり，嬉しそうに，怒ったように，悲しそうになど様々な発話スタイルで話すことができるものは少ない．その理由として，多くのテキスト音声合成システムが波形の素片を接続して音声を合成する方式を採用しているということがあげられる．波形接続型のシステムでは，音声を合成するために音声データを収集し，素片に分割し，分割した素片を格納しなければならない．波形接続型のシステムで様々な声質，発話スタイルを実現するためには，様々な声質，発話スタイルで収録された膨大な量の音声データを処理しなければならず，また合成するには膨大な量の素片を格納する記憶媒体が必要となるため，実現は非常に困難である．このような観点から，本論文では様々な話者，発話スタイルの音声を合成することができる音声合成システムの実現を目的とし，HMM に基づく音声合成システムについて述べる．

HMM に基づく音声合成システムは，(1) メルケプストラム分析合成系，(2)HMM を使った音声パラメータモデリング，(3)HMM からの音声パラメータ生成アルゴリズムの 3 つの技術から成っている．本システムは，これら 3 つの技術を用いているためいくつかの能力を持つ．一つ目に，HMM からの音声パラメータ生成アルゴリズムにより，学習した HMM から実音声に近いスペクトルパラメータ，ピッチパラメータを出力することができる．二つ目に，HMM のパラメータを適切に変換することにより様々な声質の音声を合成することができる．三つ目に，本システムは自動的に学習することができる．本論文では，上記の (1),(2),(3) の技術について述べ，音声の音韻・韻律情報を HMM の枠組で統一的にモデル化する手法を提案する．

次に，本論文では合成音声の品質を向上させるため，音声符号化手法 MELP で用いられている混合励振源モデルとポストフィルタを本システムに導入した．混合励振源モデルとポストフィルタが合成音声の品質を大幅に向上させることを実験結果で示す．

最後に，多様な話者性，発話スタイルで音声を合成するシステムの実現を目的として，話者補間手法を用いた声質変換手法を提案する．

Chapter 1

Introduction

1.1 Text-to-speech(TTS) system

Speech is the natural form of human communication, and it can enhance human-machine communication. A text-to-speech(TTS) system is one of the human-machine interface using speech. In recent years, TTS system is developed an output device of human-machine interface, and it is used in many application such as a car navigation system, information retrieval over the telephone, voice mail, a speech-to-speech translation system and so on. The goal of TTS system is to synthesize speech with natural human voice characteristics and, furthermore, with various speaker individualities and emotions (e.g., anger, sadness, joy).

The increasing availability of large speech databases makes it possible to construct TTS systems, which are referred to as data-driven or corpus-based approach, by applying statistical learning algorithms. These systems, which can be automatically trained, can generate natural and high quality synthetic speech and can reproduce voice characteristics of the original speaker.

For constructing such a system, the use of hidden Markov models (HMMs) has arisen largely. HMMs have successfully been applied to modeling the sequence of speech spectra in speech recognition systems, and the performance of HMM-based speech recognition systems have been improved by techniques which utilize the flexibility of HMMs: context-dependent modeling, dynamic feature parameters, mixtures of Gaussian densities, tying mechanism, speaker and environment adaptation techniques. HMM-based approaches to speech synthesis can be categorized as follows:

1. Transcription and segmentation of speech database [1].
2. Construction of inventory of speech segments [2]–[5].

3. Run-time selection of multiple instances of speech segments [4], [6].
4. Speech synthesis from HMMs themselves [7]–[10].

In approaches 1–3, by using a waveform concatenation algorithm, e.g., PSOLA algorithm, a high quality synthetic speech could be produced. However, to obtain various voice characteristics, large amounts of speech data are necessary, and it is difficult to collect, segment, and store the speech data. On the other hand, in approach 4, voice characteristics of synthetic speech can be changed by transforming HMM parameters appropriately. From this point of view, parameter generation algorithms [11], [12] for HMM-based speech synthesis have been proposed, and a speech synthesis system [9], [10] has been constructed using these algorithms. Actually, it has been shown that voice characteristics of synthetic speech can be changed by applying a speaker adaptation technique [13], [14] or a speaker interpolation technique [15]. The main feature of the system is the use of dynamic feature: by inclusion of dynamic coefficients in the feature vector, the dynamic coefficients of the speech parameter sequence generated in synthesis are constrained to be realistic, as defined by the parameters of the HMMs.

1.2 Proposition of this thesis

The proposed TTS system is shown in Fig. 1.1. This figure shows the training and synthesis parts of the HMM-based TTS system. In the training phase, first, spectral parameters (e.g., cepstral coefficients) and excitation parameters (e.g., fundamental frequency) are extracted from speech database. The extracted parameters are modeled by context-dependent HMMs. In the systhesis phase, a context-dependent label sequence is obtained from a input text by text analysis. A sentence HMM is constructed by concatenating context dependent HMMs according to the context-dependent label sequence. By using the parameter generation algorithm, spectral and excitation parameters are generated from the sentence HMM. Finaly, by using a synthesis filter, speech is synthesized from the generated spectral and excitation parameters.

In this thesis, it is roughly assumed that spectral and excitation parameters include phonetic and prosodic information, respectively. If these phonetic and prosodic parameters are modeled in a unified framework of HMMs, it is possible to apply speaker adaptation / interpolation techniques to phonetic and prosodic information, simultaneously, and synthesize speech with various voice characteristics such as speaker individualities and emotions. From the point of view, the phonetic and prosodic parameter modeling technique and the voice conversion technique are proposed in this thesis.

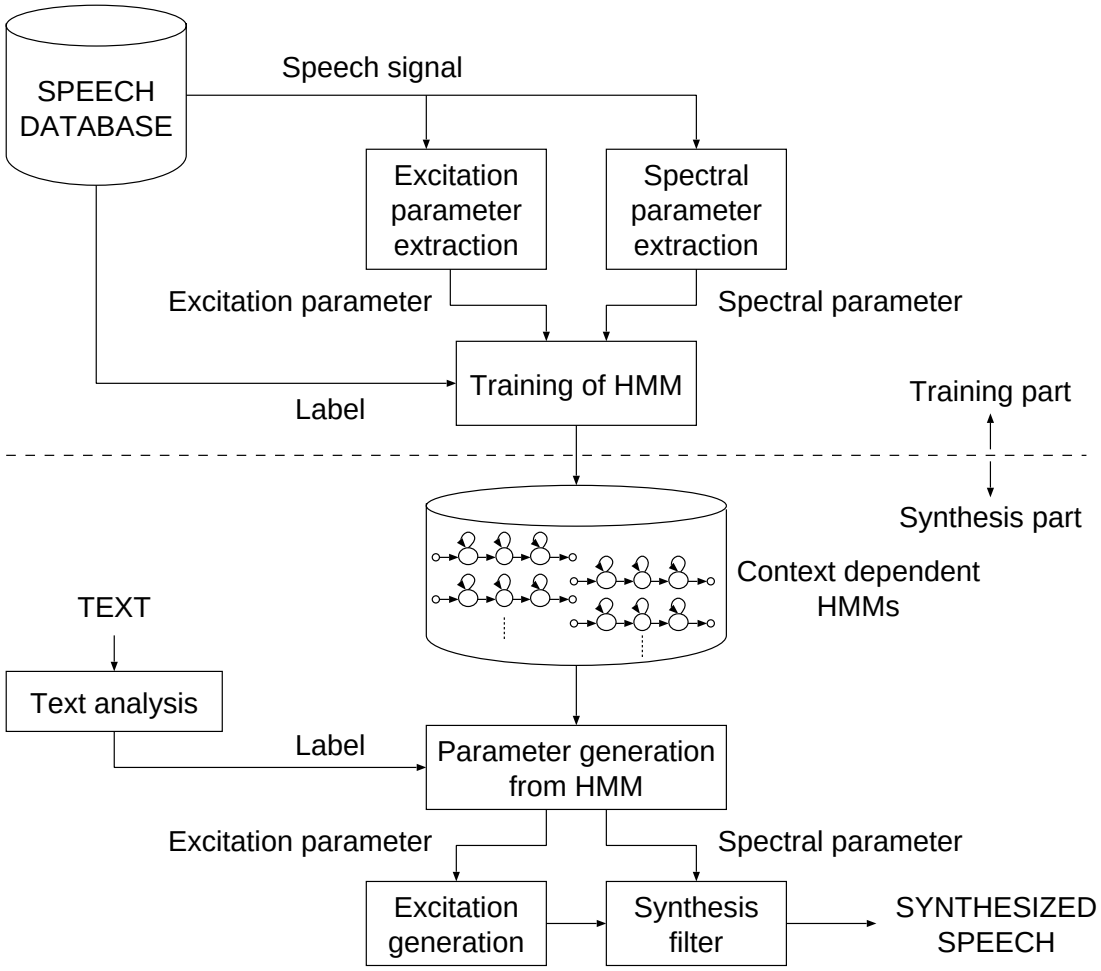


Figure 1.1: The scheme of the HMM-based TTS system.

The remainder of this thesis is organized as follows:

In **Chapters 2–4**, the following fundamental techniques of the HMM-based speech synthesis are described:

- Mel-cepstral analysis and synthesis technique (**Chapter 2**)
- Speech parameter modeling based on HMM (**Chapter 3**)
- Speech parameter generation from HMM (**Chapter 4**)

Chapter 5 presents construction of the proposed HMM-based TTS system in which spectrum, fundamental frequency (F0), duration are modeled by HMM simultaneously, and **Chapter 6** discribes how to synthesize speech. In **Chapter 7**, mixed excitation model is incorporated into the proposed TTS system in order to improve

quality of synthesized speech. **Chapter 8** presents how to synthesize speech with various voice characteristics, appling speaker interpolation technique to HMM-based TTS. Finally **Chapter 9** draws overall conclusions and describes possible future works.

1.3 Original contributions

This thesis describes new approaches to synthesize speech with natural human voice characteristics and with various voice characteristics such as speaker individuality and emotion. The major original contributions are as follows:

- Speech parameter generation using multi-mixture HMM.
- Duration modeling for the HMM-based TTS system.
- Simultaneous modeling of spectrum, F0 and duration.
- Training of context dependent HMM using MDL principle.
- F0 parameter generation using dynamic features.
- Automatic training of the HMM-based TTS system.
- Improvement of the quality of the synthesized speech by incorporating the mixed excitation model and postfilter into the HMM-based TTS system.
- Voice conversion using a speaker interpolation technique.

Chapter 2

Mel-cepstral Analysis and Synthesis Technique

The proposed TTS system is based on source-filter model. In order to construct the system, first, it is necessary to extract feature parameters, which describe the vocal tract, from speech database for training. For all the work in this thesis, the mel-cepstral analysis [16] is used for spectral estimation. This chapter describes the mel-cepstral analysis, how feature parameters, i.e., mel-cepstral coefficients, are extracted speech signals and how speech is synthesized from the mel-cepstral coefficients.

2.1 Source-filter model

To treat a speech waveform mathematically, source-filter model is generally used to represent sampled speech signals, as shown in 2.1. The transfer function $H(z)$ models the structure of vocal tract. The excitation source is chosen by a switch which controls the voiced/unvoiced character of the speech. The excitation signal is modeled as either a periodic pulse train for voiced speech, or a random noise sequence for unvoiced speech. To produce speech signal $x(n)$, the parameters of the model must change with time. The excitation signal $e(n)$ is filtered by a time-varying linear system $H(z)$ to generate speech signals $x(n)$.

The speech $x(n)$ can be computed from the excitation $e(n)$ and the impulse response $h(n)$ of the vocal tract using the convolution sum expression

$$x(n) = h(n) * e(n) \tag{2.1}$$

where the symbol $*$ stands for discrete convolution. The details of digital signal processing and speech processing are given in Ref. [17]

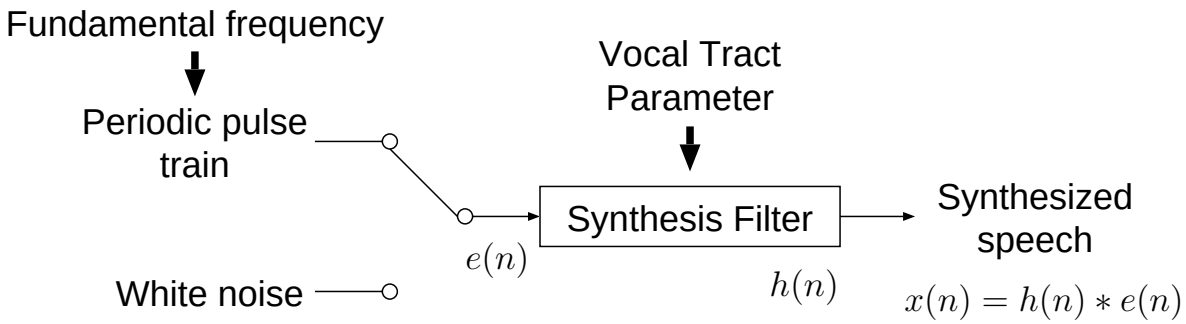


Figure 2.1: Source-filter model.

2.2 Mel-cepstral analysis

In mel-cepstral analysis[16], the model spectrum $H(e^{jw})$ is represented by the M-th order mel-cepstral coefficients $\tilde{c}(m)$ as follows:

$$H(z) = \exp \sum_{m=0}^{M} \tilde{c}(m)\tilde{z}^{-m}, \tag{2.2}$$

where

$$\tilde{z}^{-1} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}, \quad |a| < 1. \tag{2.3}$$

The phase characteristic of the all-pass transfer function $\tilde{z}^{-1} = e^{-j\tilde{\omega}}$ is given by

$$\tilde{\omega} = \tan^{-1} \frac{(1-\alpha^2)\sin\omega}{(1+\alpha^2)\cos\omega - 2\alpha}. \tag{2.4}$$

For example for a sampling frequency of 16kHz, $\tilde{\omega}$ is a good approximation to the mel scale based on subjective pitch evaluations when $\alpha = 0.42$(Tab. 2.1).

To obtain an unbiased estimate, we use the following criterion [18] and minimize it with respect to $\tilde{c}(m)_{m=0}^{M}$

$$E = \frac{1}{2\pi} \int_{-\pi}^{\pi} \exp R(\omega) - R(\omega) - 1 d\omega, \tag{2.5}$$

where

$$R(\omega) = \log I_N(\omega) - log \left| H(e^{j\omega}) \right|^2, \tag{2.6}$$

and $I_N(\omega)$ is the modified periodogram of a weakly stationary process $x(n)$ with a time window of length N. To take the gain factor K outside from $H(z)$, we rewrite Eq.(2.2) as:

$$H(z) = \exp \sum_{m=0}^{M} b(m)\Phi_m(z) = K \cdot D(z), \tag{2.7}$$

Table 2.1: Examples of α for approximating auditory frequency scales.

Sampling frequency	Mel scale	Bark scale
8kHz	0.31	0.42
10kHz	0.35	0.47
12kHz	0.37	0.50
16kHz	0.42	0.55

where

$$K = \exp b(0), \tag{2.8}$$

$$D(z) = \exp \sum_{m=0}^{M} b(m)\Phi_m(z), \tag{2.9}$$

and

$$b(m) = \begin{cases} c(m) & m = M \\ c(m) - \alpha b(m+1) & 0 \le m < M \end{cases} \tag{2.10}$$

$$\Phi_m(z) = \begin{cases} 1 & m = 0 \\ \dfrac{(1-\alpha^2)z^{-1}}{1-\alpha z^{-1}} \tilde{z}^{-(m-1)} & m \ge 1 \end{cases} \tag{2.11}$$

Since $H(z)$ is a minimum phase system, we can show that the minimization of E with respect to $\tilde{c}(m)_{m=0}^{M}$ is equivalent to that of

$$\varepsilon = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{I_N(\omega)}{|D(e^{j\omega})|} d\omega, \tag{2.12}$$

with respect to

$$\boldsymbol{b} = [b(1), b(2), \cdots, b(M)]^T. \tag{2.13}$$

The gain factor K that minimizes E is obtained by setting $\frac{\partial E}{\partial K} = 0$:

$$K = \sqrt{\varepsilon_{min}} \tag{2.14}$$

where ε_{min} is the minimized value of ε.

2.3 Synthesis filter

The synthesis filter $D(z)$ of Eq.(2.9) is not a rational function and, therefore, it cannot be realized directly. Howerver, using Mel Log Spectrum Approximation filter

Table 2.2: Coefficients of $R_4(\omega)$.

l	$A_{4,l}$
1	4.999273×10^{-1}
2	1.067005×10^{-1}
3	1.170221×10^{-2}
4	5.656279×10^{-4}

(MLSA filter) [19], the synthesis filter $D(z)$ can be approximated with sufficient accuracy and becomes minimum phase IIR system. The complex exponential function $\exp\omega$ is qpproximated by a rational function

$$\begin{aligned} \exp\omega &\simeq R_L(F(z)) \\ &= \frac{1+\sum_{l=1}^{L} A_{L,l}\omega^l}{1+\sum_{l=1}^{L} A_{L,l}(-\omega)^l}. \end{aligned} \tag{2.15}$$

Thus $D(z)$ is approximated as follows:

$$R_L(F(z)) \simeq exp(F(z)) = D(z) \tag{2.16}$$

where $F(z)$ is defined by

$$F(z) = \sum_{m=1}^{M} b(m)\Phi_m(z). \tag{2.17}$$

The filter structure of $F(z)$ is shown in Fig. 2.2(a). Figure 2.2(b) shows the block diagram of the MLSA filter $R_L(F(z))$ for the case of $L = 4$.

When we use the coefficients $A_{4,l}$ show in Tab. 2.2, $R_4(F(z))$ is stable and becomes a minimum phase system under the condition

$$|F(e^{j\omega})| \leq 6.2. \tag{2.18}$$

Further more, we can show that the approximation error $|\log D(e^{j\omega}) - \log R_4(F(e^{j\omega}))|$ does not exceed 0.24dB[20] under the condition

$$|F(e^{j\omega})| \leq 4.5. \tag{2.19}$$

When $F(z)$ is expressed as

$$F(z) = F_1(z) + F_2(z) \tag{2.20}$$

the exponential transfer function is approximated in a cascade form

$$\begin{aligned} D(z) &= \exp F(z) \\ &= \exp F_1(z) \cdot \exp F_2(z) \\ &\simeq R_L(F_1(z)) \cdot R_L(F_2(z)) \end{aligned} \tag{2.21}$$

as shown in Fig. 2.2(c). If

$$\max_\omega |F_1(e^{j\omega})|, \max_\omega |F_2(e^{j\omega})| < \max_\omega |F(e^{j\omega})|, \tag{2.22}$$

it is expected that $R_L(F_1(e^{j\omega})) \cdot R_L(F_2(e^{j\omega}))$ approximates $D(e^{j\omega})$ more accurately than $R_L(F(e^{j\omega}))$.

In the following experiments, we let

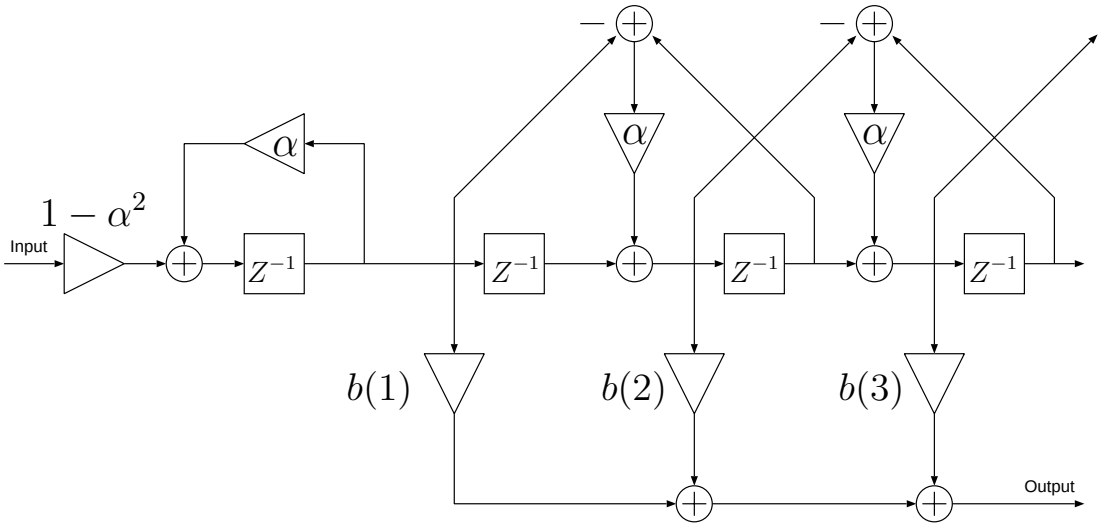
$$F_1(z) = b(1)\Phi_1(z) \tag{2.23}$$

$$F_2(z) = \sum_{m=2}^{M} b(m)\Phi_m(z). \tag{2.24}$$

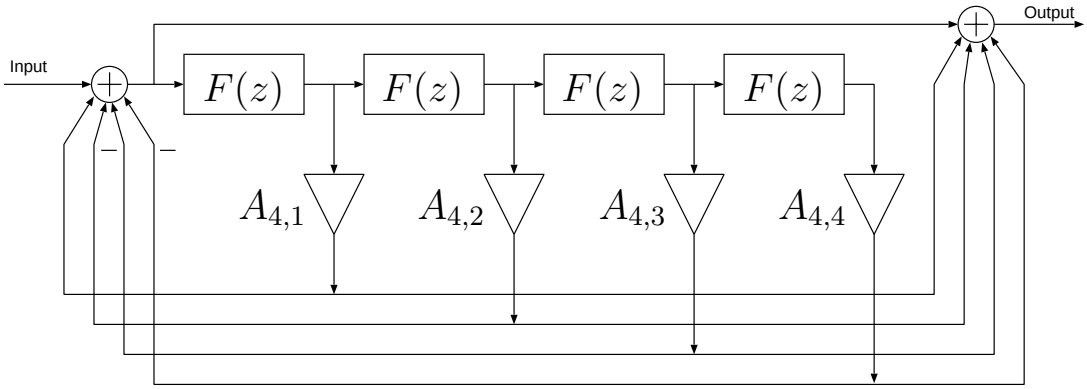
Since we empirically found that

$$\max_\omega |F_1(e^{j\omega})|, \max_\omega |F_2(e^{j\omega})| < 4.5 \tag{2.25}$$

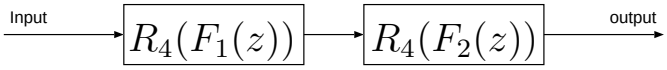
for speech sounds, $R_L(F_1(z)) \cdot R_L(F_2(z))$ approximates the exponential transfer function $D(z)$ with sufficient accuracy and becomes a stable system.



(a) Basic filter $F(z)$



(b) $R_L(F(z)) \simeq D(z), \quad L = 4$



(c) Two stage cascade structure

Figure 2.2: Implementation of Synthesis filter $D(z)$.

Chapter 3

Speech Parameter Modeling Based on HMM

The most successful and widely used acoustic models in recent years have been Hidden Markov Models (HMMs). Practically all major speech recognition systems are generally implemented using HMM. This chapter describes how to model spectral and excitation parameters in a framework of HMM.

3.1 Spectral parameter modeling

3.1.1 Continuous density HMM

In this thesis, a continuous density HMM is used for the vocal tract modeling in the same way as speech recognition systems. The continuous density Markov model is a finite state machine which makes one state transition at each time unit (i.e, frame). First, a decision is made to which state to succeed (including the state itself). Then an output vector is generated according to the probability density function (pdf) for the current state. An HMM is a doubly stochastic random process, modeling state transition probabilities between states and output probabilities at each state.

One way of interpreting HMMs is to view each state as a model of a segment of speech. Figure 3.1 shows an example of representation of a speech utterance using a N-state left-to-right HMM where each state is modeled by a multi-mixture Gaussian model. Assume that this utterance(typically parameterized by speech analysis as the D-dimensional observation vector $\mathbf{o}_t$) is divided into N segments d_i which are represented by the states S_i. The transition probability a_{ij} defines the probability of moving from state i to state j and satisfies $a_{ii} + a_{ij} = 1$. Then, each state can be

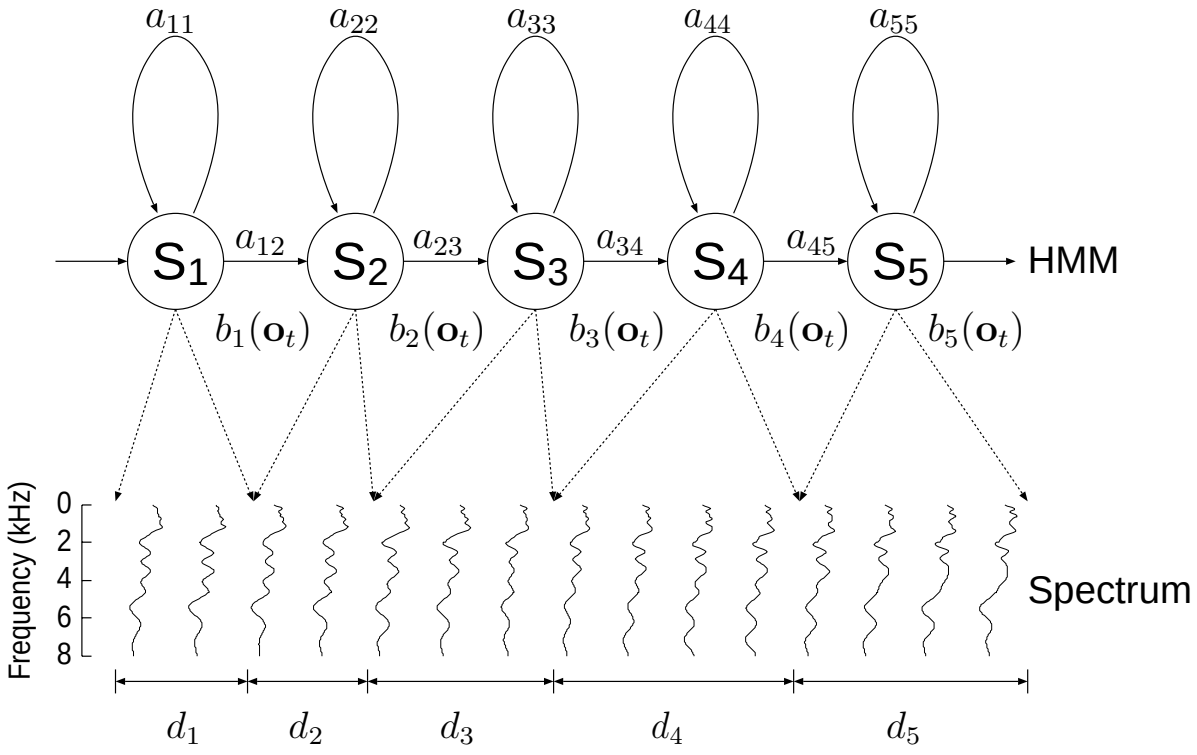


Figure 3.1: Representation of a speech utterance using a five-state HMM.

modeled by a M-mixtures Gaussian density function:

$$\begin{aligned} b_j(\mathbf{o}_t) &= \sum_{k=1}^{M} c_{jk}\mathcal{N}(\mathbf{o}_t, \mu_{jk}, \mathbf{\Sigma}_{jk}) \\ &= \sum_{k=1}^{M} c_{jk} \frac{1}{(2\pi)^{\frac{D}{2}}|\mathbf{\Sigma}_{jk}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\mathbf{o}_t - \mu_{jk})^T \mathbf{\Sigma}_{jk}^{-1}(\mathbf{o}_t - \mu_{jk})\right\}, \end{aligned} \tag{3.1}$$

where c_{jk}, μ_{jk} and Σ_{jk} are the mixture coefficient, D-dimensional mean vector and $D \times D$ covariance matrix(full covariance matrix) for the k-th mixture component in the j-th state, respectively. This covariance can be restricted to the diagonal elements (diagonal covariance matrix) when the elements of the feature vector are assumed to be independent. $|\mathbf{\Sigma}_{jk}|$ is the determinant of $\mathbf{\Sigma}_{jk}$, and $\mathbf{\Sigma}_{jk}^{-1}$ is the inverse of $\mathbf{\Sigma}_{jk}$. The mixture gains c_{jk} satisfy the stochastic constraint

$$\sum_{k=1}^{M} c_{jk} = 1, \quad 1 \le j \le N \tag{3.2}$$

$$c_{jk} \ge 0, \quad 1 \le j \le N, 1 \le k \le M \tag{3.3}$$

so that the pdf is properly normalized, i.e.,

$$\int_{-\infty}^{\infty} b_j(\mathbf{o}) d\mathbf{o}, \quad 1 \leq j \leq N \tag{3.4}$$

Since the pdf of Eq. (3.1) can be used to approximate, arbitrarily closely, any finite, continuous density function, it can be applied to a wide range of problems and is widely used for acoustic modeling.

For convenience, to indicate the complete parameter set of the model, we use the compact notation

$$\lambda = (\mathbf{A}, \mathbf{B}, \pi), \tag{3.5}$$

where $A = \{a_{ij}\}$, $B = \{b_j(\mathbf{o})\}$ and $\pi = \{\pi_i\}$. π_i is the initial state distribution of state i, and it have the property

$$\pi = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases} \tag{3.6}$$

in the left-to-right model.

3.1.2 Probability calculation

For calculation of $P(\mathbf{O}|\lambda)$, which is the probability of the observation sequence $\mathbf{O} = (\mathbf{o}_1, \mathbf{o}_2, \cdots, \mathbf{o}_T)$ given the model λ, forward-backward algorithm is generally used. If we calculate $P(\mathbf{O}|\lambda)$ directly without this algorithm, it requires on the order of $2TN^2$ calculation. On the other hand, calculation using forward-backward algorithm requires on the order of N^2T calculations, and it is computationally feasible. In the following part, forward-backward algorithm is described.

The forward algorithm

Consider the forward variable $\alpha_t(i)$ defined as

$$\alpha_t(i) = P(\mathbf{o}_1, \mathbf{o}_2, \cdots, \mathbf{o}_t, q_t = i|\lambda) \tag{3.7}$$

that is, the probability of the partial observation sequence from 1 to t and state i at time t, given the model λ. We can solve for $\alpha_t(i)$ inductively, as follows:

1. Initialization

$$\alpha_1(i) = \pi_i b_i(\mathbf{o}_1), \quad 1 \leq ileqN. \tag{3.8}$$

2. Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(\mathbf{o}_{t+1}), \qquad \begin{array}{l} 1 \le t \le T-1 \\ 1 \le j \le N \end{array} \tag{3.9}$$

3. Termination

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \tag{3.10}$$

The backward algorithm

In the same way as forward algorithm, consider the backward variable $\beta_t(i)$ defined as

$$\beta_t(i) = P(\mathbf{o}_t + 1, \mathbf{o}_t + 2, \cdots, \mathbf{o}_T | q_t = i, \lambda) \tag{3.11}$$

that is, the probability of the partial observation sequence from t to T, given state i at time t and the model λ. We can solve for $\beta_t(i)$ inductively, as follows:

1. Initialization

$$\beta_T(i) = 1, \quad 1 \le i \le N. \tag{3.12}$$

2. Induction

$$\beta_t(i) = \sum_{i=1}^{N} a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j), \qquad \begin{array}{l} t = T-1, T-2, \cdots, 1 \\ 1 \le i \le N \end{array} \tag{3.13}$$

3. Termination

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \beta_1(i). \tag{3.14}$$

The forward-backward probability calculation is based on the trellis structure shown in Fig. 3.2. In this figure, the x-axis and y-axis represent observation sequence and states of Markov model, respectively. On the trellis, all the possible state sequence will remerge into these N nodes no matter how long the observation sequence. In the case of the forward algorithm, at times $t = 1$, we need to calculate values of $\alpha_1(i)$, $1 \le i \le N$. At times $t = 2, 3, \cdots, T$, we need only calculate values of $\alpha_t(j)$, $1 \le j \le N$, where each calculation involves only the N previous values of $\alpha_{t-1}(i)$ because each of the N grid points can be reached from only the N grid points at the previous time slot. As the result, the forward-backward algrorithm can reduce order of probability calculation.

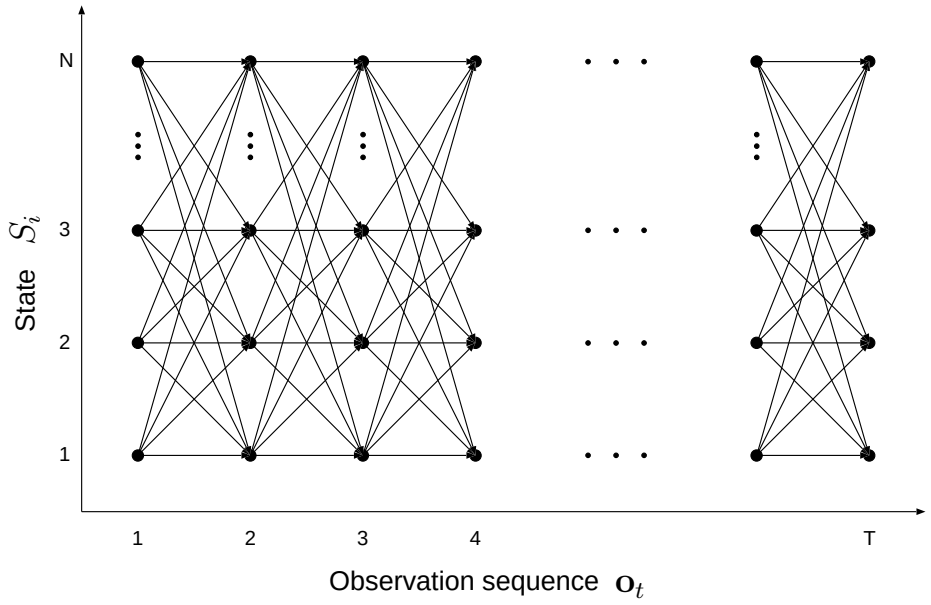


Figure 3.2: Implementation of the computation using forward-backward algorithm in terms of a trellis of observation t and state i.

3.1.3 Parameter estimation of continuous density HMM

It is difficult to determine a method to adjust the model parameters $(\mathbf{A}, \mathbf{B}, \pi)$ to satisfy a certain optimization criterion. There is no known way to analytical solve for the model parameter set that maximizes the probability of the observation sequence in a closed form. We can, however, choose $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$ such that its likelihood, $P(\mathbf{O}|\lambda)$, is locally maximized using an iterative procedure such as the Baum-Welch method(also known as the EM(expectation-maximization method))[21], [22].

To describe the procedure for reestimation of HMM parameters, first, the probability of being in state i at time t, and state j at time $t+1$, given the model and the observation sequence, is defined as follows:

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j|\mathbf{O}, \lambda), \tag{3.15}$$

From the definitions of the forward and backward variables, $\xi_t(i,j)$ is writen in the form

$$\begin{aligned} \xi_t(i,j) &= \frac{P(q_t = i, q_{t+1} = j|\mathbf{O}, \lambda)}{P(\mathbf{O}|\lambda)} \\ &= \frac{\alpha_t(i)a_{ij}b_j(\mathbf{o}_{t+1}\beta_{t+1}(j))}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(\mathbf{o}_{t+1}\beta_{t+1}(j))} \end{aligned} \tag{3.16}$$

Using $\xi_t(i,j)$, the probability of being in state i at time t, given the entire observation and the model, is represented by

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j). \tag{3.17}$$

Q-function

The reestimation formulas can be derived directly by maximizing Baum's auxiliary funciton

$$Q(\lambda', \lambda) = \sum_{\mathbf{q}} P(\mathbf{O}, \mathbf{q}|\lambda') \log P(\mathbf{O}, \mathbf{q}|\lambda) \tag{3.18}$$

over λ. Because

$$Q(\lambda', \lambda) \geq Q(\lambda', \lambda) \Rightarrow P(\mathbf{O}, \mathbf{q}|\lambda') \geq P(\mathbf{O}, \mathbf{q}|\lambda) \tag{3.19}$$

We can maximize the function $Q(\lambda', \lambda)$ over λ to improve λ' in the sense of increasing the likelihood $P(\mathbf{O}, \mathbf{q}|\lambda)$.

Maximization of Q-function

For given observation sequence $\mathbf{O}$ and model λ', we derive parameters of λ which maximize $Q(\lambda', \lambda)$. $P(\mathbf{O}, \mathbf{q}|\lambda)$ can be written as

$$P(\mathbf{O}, \mathbf{q}|\lambda) = \pi_{q_0} \prod_{t=1}^{T} a_{q_{t-1}q_t} b_{q_t}(\mathbf{o}_t) \tag{3.20}$$

$$\log P(\mathbf{O}, \mathbf{q}|\lambda) = \log \pi_{q_0} + \sum_{t=1}^{T} \log a_{q_{t-1}q_t} + \sum_{t=1}^{T} \log b_{q_t}(\mathbf{o}_t) \tag{3.21}$$

Hence Q-function (3.18) can be written as

$$\begin{aligned} Q(\lambda', \lambda) &= Q_\pi(\lambda', \mathbf{pi}) \\ &+ \sum_{t=1}^{T} Q_{a_i}(\lambda', \mathbf{a}_i) \\ &+ \sum_{t=1}^{T} Q_{b_i}(\lambda', \mathbf{b}_i) \\ &= \sum_{i=1}^{N} P(\mathbf{O}, q_0 = i|\lambda') \log \pi_i \\ &+ \sum_{j=1}^{N} \sum_{t=1}^{T} P(\mathbf{O}, q_{t-1} = i, q_t = j|\lambda') \log a_{ij} \\ &+ \sum_{t=1}^{T} P(\mathbf{O}, q_t = i|\lambda') \log b_i(\mathbf{o}_t) \end{aligned} \tag{3.22}$$

where

$$\pi = [\pi_1, \pi_2, \cdots, \pi_N] \tag{3.23}$$

$$\mathbf{a}_i = [a_{i1}, a_{i2}, \cdots, a_{iN}], \tag{3.24}$$

and $\mathbf{b}_i$ is the parameter vector that defines $b_i(\cdot)$. The parameter set λ which maximizes (3.22), subject to the stochastic constraints

$$\sum_{j=1}^{N} \pi_j = 1, \tag{3.25}$$

$$\sum_{j=1}^{N} a_{ij} = 1, \tag{3.26}$$

$$\sum_{k=1}^{M} c_{jk} = 1, \tag{3.27}$$

$$\int_{-\infty}^{\infty} b_j(\mathbf{o}) d\mathbf{o} = 1, \tag{3.28}$$

can be derived as

$$\pi_i = \frac{\alpha_0(i)\beta_0(i)}{\sum_{j=1}^{N} \alpha_T(j)} = \gamma_0(i) \tag{3.29}$$

$$a_{ij} = \frac{\sum_{t=1}^{T} \alpha_{t-1}(i) a_{ij} b_j(\mathbf{o}_t) \beta_t(j)}{\sum_{t=1}^{T} \alpha_{t-1}(i)\beta_{t-1}(i)} = \frac{\sum_{t=1}^{T} \xi_{t-1}(i,j)}{\sum_{t=1}^{T} \gamma_{t-1}(i)} \tag{3.30}$$

The reestimation formulas for the coefficients of the mixture density, i.e, c_{jk}, μ_{jk} and $\mathbf{\Sigma}_{jk}$ are of the form

$$c_{ij} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \gamma_t(j,k)} \tag{3.31}$$

$$\mu_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot \mathbf{o}_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{3.32}$$

$$\mathbf{\Sigma}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot (\mathbf{o}_t - \mu_{jk})(\mathbf{o}_t - \mu_{jk})'}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{3.33}$$

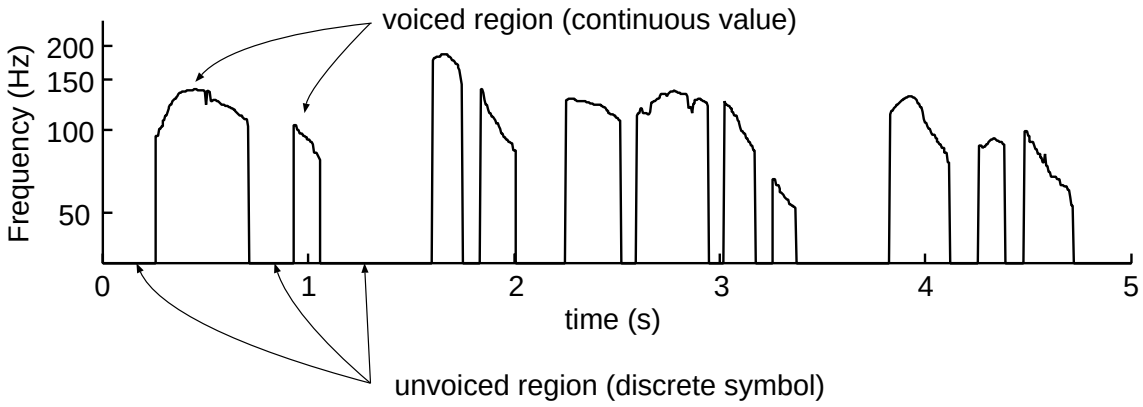


Figure 3.3: Example of F0 pattern.

(3.34)

where $\gamma_t(j,k)$ is the probability of being in state j at time t with the kth mixture component accounting for $\mathbf{o}_t$, i.e.,

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jk}\mathcal{N}(\mathbf{o}_t,\mu_{jk},\mathbf{\Sigma}_{jk})}{\sum_{m=1}^{M}\mathcal{N}(\mathbf{o}_t,\mu_{jk},\mathbf{\Sigma}_{jk})}\right]. \tag{3.35}$$

3.2 F0 parameter modeling

The F0 pattern is composed of continuous values in the "voiced" region and a discrete symbol in the "unvoiced" region (Fig.3.3). Therefore, it is difficult to apply the discrete or continuous HMMs to F0 pattern modeling. Several methods have been investigated [23] for handling the unvoiced region: (i) replacing each "unvoiced" symbol by a random vector generated from a probability density function (pdf) with a large variance and then modeling the random vectors explicitly in the continuous HMMs [24], (ii) modeling the "unvoiced" symbols explicitly in the continuous HMMs by replacing "unvoiced" symbol by 0 and adding an extra pdf for the "unvoiced" symbol to each mixture, (iii) assuming that F0 values is always exist but they cannot observed in the unvoiced region and applying the EM algorithm [25]. In this section, A kind of HMM for F0 pattern modeling, in which the state output probabilities are defined by multi-space probability distributions (MSDs), is described.

3.2.1 Multi-Space Probability Distribution

We consider a sample space Ω shown in Fig. 3.4, which consists of G spaces:

$$\Omega = \bigcup_{g=1}^{G} \Omega_g \tag{3.36}$$

where Ω_g is an n_g-dimensional real space R^{n_g}, and specified by space index g. Each space Ω_g has its probability w_g, i.e., $P(\Omega_g) = w_g$, where $\sum_{g=1}^{G} w_g = 1$. If $n_g > 0$, each space has a probability density function $\mathcal{N}_g(\boldsymbol{x})$, $\boldsymbol{x} \in R^{n_g}$, where $\int_{R^{n_g}} \mathcal{N}_g(\boldsymbol{x}) d\boldsymbol{x} = 1$. We assume that Ω_g contains only one sample point if $n_g = 0$. Accordingly, letting $P(E)$ be the probability distribution, we have

$$P(\Omega) = \sum_{g=1}^{G} P(\Omega_g) = \sum_{g=1}^{G} w_g \int_{R^{n_g}} \mathcal{N}_g(\boldsymbol{x}) d\boldsymbol{x} = 1. \tag{3.37}$$

It is noted that, although $\mathcal{N}_g(\boldsymbol{x})$ does not exist for $n_g = 0$ since Ω_g contains only one sample point, for simplicity of notation, we define as $\mathcal{N}_g(\boldsymbol{x}) \equiv 1$ for $n_g = 0$.

Each event E, which will be considered in this thesis, is represented by a random variable $\boldsymbol{o}$ which consists of a continuous random variable $\boldsymbol{x} \in R^n$ and a set of space indices X, that is,

$$\boldsymbol{o} = (\boldsymbol{x}, X) \tag{3.38}$$

where all spaces specified by X are n-dimensional. The observation probability of $\boldsymbol{o}$ is defined by

$$b(\boldsymbol{o}) = \sum_{g \in S(\boldsymbol{o})} w_g \mathcal{N}_g(V(\boldsymbol{o})) \tag{3.39}$$

where

$$V(\boldsymbol{o}) = \boldsymbol{x}, \qquad S(\boldsymbol{o}) = X. \tag{3.40}$$

Some examples of observations are shown in Fig. 3.4. An observation $\boldsymbol{o}_1$ consists of three-dimensional vector $\boldsymbol{x}_1 \in R^3$ and a set of space indices $X_1 = \{1, 2, G\}$. Thus the random variable $\boldsymbol{x}$ is drawn from one of three spaces Ω_1, Ω_2, $\Omega_G \in R^3$, and its probability density function is given by $w_1\mathcal{N}_1(\boldsymbol{x}) + w_2\mathcal{N}_2(\boldsymbol{x}) + w_G\mathcal{N}_G(\boldsymbol{x})$.

The probability distribution defined in the above, which will be referred to as *multi-space probability distribution* (MSD) in this thesis, is the same as the discrete distribution and the continuous distribution when $n_g \equiv 0$ and $n_g \equiv m > 0$, respectively. Further, if $S(\boldsymbol{o}) \equiv \{1, 2, \ldots, G\}$, the continuous distribution is represented by a G-mixture probability density function. Thus multi-space probability distribution is more general than either discrete or continuous distributions.

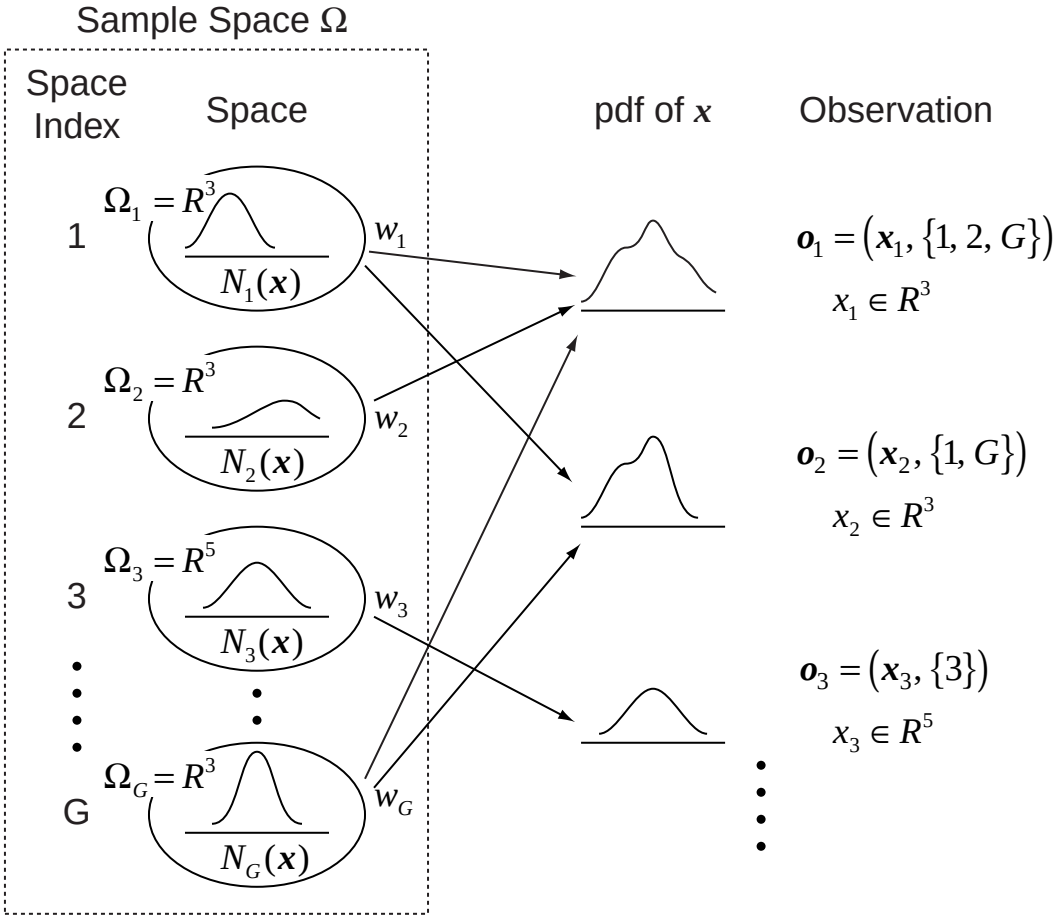


Figure 3.4: Multi-space probability distribution and observations.

3.2.2 Multi-space distribution HMM

The output probability in each state of MSD-HMM is given by the multi-space probability distribution defined in the previous section. An N-state MSD-HMM λ is specified by initial state probability distribution $\pi = \{\pi_j\}_{j=1}^N$, the state transition probability distribution $A = \{a_{ij}\}_{i,j=1}^N$, and state output probability distribution $B = \{b_i(\cdot)\}_{i=1}^N$, where

$$b_i(\boldsymbol{o}) = \sum_{g \in S(\boldsymbol{o})} w_{ig} \ \mathcal{N}_{ig}(V(\boldsymbol{o})), \quad i = 1, 2, \ldots, N. \tag{3.41}$$

As shown in Fig. 3.5, each state i has G probability density functions $\mathcal{N}_{i1}(\cdot)$, $\mathcal{N}_{i2}(\cdot)$, …, $\mathcal{N}_{iG}(\cdot)$, and their weights w_{i1}, w_{i2}, …, w_{iG}.

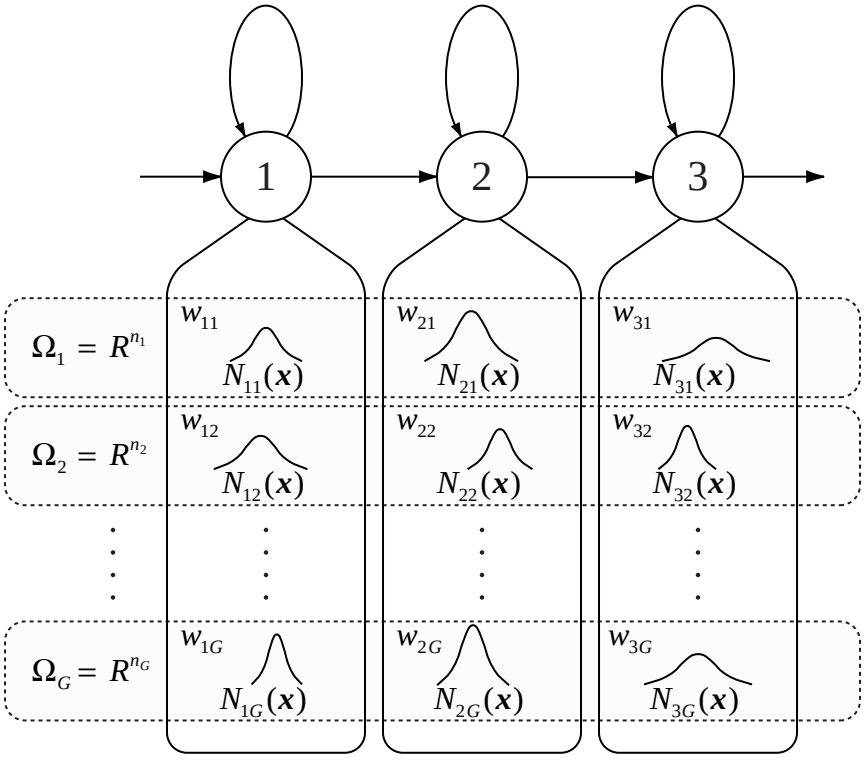


Figure 3.5: An HMM based on multi-space probability distribution.

Observation probability of $\boldsymbol{O} = \{\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T\}$ is written as

$$\begin{aligned} P(\boldsymbol{O}|\lambda) &= \sum_{\text{all } \boldsymbol{q}} \prod_{t=1}^{T} a_{q_{t-1}q_t}\, b_{q_t}(\boldsymbol{o}_t) \\ &= \sum_{\text{all } \boldsymbol{q},\boldsymbol{l}} \prod_{t=1}^{T} a_{q_{t-1}q_t}\, w_{q_t l_t}\, \mathcal{N}_{q_t l_t}(V(\boldsymbol{o}_t)) \end{aligned} \tag{3.42}$$

where $\boldsymbol{q} = \{q_1, q_2, \ldots, q_T\}$ is a possible state sequence, $\boldsymbol{l} = \{l_1, l_2, \ldots, l_T\} \in \{S(\boldsymbol{o}_1) \times S(\boldsymbol{o}_2) \times \ldots \times S(\boldsymbol{o}_T)\}$ is a sequence of space indices which is possible for the observation sequence $\boldsymbol{O}$, and $a_{q_0 j}$ denotes π_j.

The forward and backward variables:

$$\alpha_t(i) = P(\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_t, q_t = i|\lambda) \tag{3.43}$$

$$\beta_t(i) = P(\boldsymbol{o}_{t+1}, \boldsymbol{o}_{t+2}, \ldots, \boldsymbol{o}_T|q_t = i, \lambda) \tag{3.44}$$

can be calculated with the forward-backward inductive procedure in a manner similar to the conventional HMMs. According to the definitions, (3.42) can be calculated as

$$P(\boldsymbol{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i) = \sum_{i=1}^{N} \beta_1(i). \tag{3.45}$$

The forward and backward variables are also used for calculating the reestimation formulas derived in the the next section

3.2.3 Reestimation algorithm for MSD-HMM training

For a given observation sequence $\boldsymbol{O}$ and a particular choice of MSD-HMM, the objective in maximum likelihood estimation is to maximize the observation likelihood $P(\boldsymbol{O}|\lambda)$ given by (3.42), over all parameters in λ. In a manner similar to [21], [22], we derive reestimation formulas for the maximum likelihood estimation of MSD-HMM.

Q-function

An auxiliary function $Q(\lambda', \lambda)$ of current parameters λ' and new parameter λ is defined as follows:

$$Q(\lambda', \lambda) = \sum_{\text{all } \boldsymbol{q},\boldsymbol{l}} P(\boldsymbol{O}, \boldsymbol{q}, \boldsymbol{l}|\lambda') \log P(\boldsymbol{O}, \boldsymbol{q}, \boldsymbol{l}|\lambda) \tag{3.46}$$

In the following, we assume $\mathcal{N}_{ig}(\cdot)$ to be the Gaussian density with mean vector $\boldsymbol{\mu}_{ig}$ and covariance matrix $\boldsymbol{\Sigma}_{ig}$.

Theorem 1

$$Q(\lambda', \lambda) \geq Q(\lambda', \lambda') \rightarrow P(\boldsymbol{O}, \lambda) \geq P(\boldsymbol{O}, \lambda')$$

Theorem 2 *If, for each space Ω_g, there are among $V(\boldsymbol{o}_1)$, $V(\boldsymbol{o}_2)$, ..., $V(\boldsymbol{o}_T)$, n_g+1 observations $g \in S(o_t)$, any n_g of which are linearly independent, $Q(\lambda', \lambda)$ has a unique global maximum as a function of λ, and this maximum is the one and only critical point.*

Theorem 3 *A parameter set λ is a critical point of the likelihood $P(\boldsymbol{O}|\lambda)$ if and only if it is a critical point of the Q-function.*

We define the parameter reestimates to be those which maximize $Q(\lambda', \lambda)$ as a function of λ, λ' being the latest estimates. Because of the above theorems, the sequence of resetimates obtained in this way produce a monotonic increase in the likelihood unless λ is a critical point of the likelihood.

Maximization of Q-function

For given observation sequence $\boldsymbol{O}$ and model λ', we derive parameters of λ which maximize $Q(\lambda', \lambda)$. From (3.42), $\log P(\boldsymbol{O}, \boldsymbol{q}, \boldsymbol{l}|\lambda)$ can be written as

$$\begin{aligned}
&\log P(\boldsymbol{O}, \boldsymbol{q}, \boldsymbol{l}|\lambda) \\
&= \sum_{t=1}^{T} \Big(\log a_{q_{t-1}q_t} + \log w_{q_t l_t} + \log \mathcal{N}_{q_t l_t}(V(\boldsymbol{o}_t)) \Big) .
\end{aligned} \tag{3.47}$$

Hence Q-function (3.46) can be written as

$$\begin{aligned}
Q(\lambda', \lambda) &= \sum_{i=1}^{N} P(\boldsymbol{O}, q_1 = i|\lambda') \log \pi_i \\
&\quad + \sum_{i,j=1}^{N} \sum_{t=1}^{T-1} P(\boldsymbol{O}, q_t = i, q_{t+1} = j|\lambda') \log a_{ij} \\
&\quad + \sum_{i=1}^{N} \sum_{g=1}^{G} \sum_{t \in T(\boldsymbol{O},g)} P(\boldsymbol{O}, q_t = i, l_t = g|\lambda') \log w_{ig} \\
&+ \sum_{i=1}^{N} \sum_{g=1}^{G} \sum_{t \in T(\boldsymbol{O},g)} P(\boldsymbol{O}, q_t = i, l_t = g|\lambda') \log \mathcal{N}_{ig}(V(\boldsymbol{o}_t))
\end{aligned} \tag{3.48}$$

where

$$T(\boldsymbol{O}, g) = \{t \,|\, g \in S(\boldsymbol{o}_t)\}. \tag{3.49}$$

The parameter set $\lambda = (\pi, A, B)$ which maximizes (3.48), subject to the stochastic constraints $\sum_{i=1}^{N} \pi_i = 1$, $\sum_{j=1}^{N} a_{ij} = 1$ and $\sum_{g=1}^{G} w_g = 1$, can be derived as

$$\pi_i = \sum_{g \in S(\boldsymbol{o}_1)} \gamma'_1(i, g) \tag{3.50}$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi'_t(i, j)}{\sum_{t=1}^{T-1} \sum_{g \in S(\boldsymbol{o}_t)} \gamma'_t(i, g)} \tag{3.51}$$

$$w_{ig} = \frac{\sum_{t \in T(\boldsymbol{O},g)} \gamma'_t(i, g)}{\sum_{h=1}^{G} \sum_{t \in T(\boldsymbol{O},h)} \gamma'_t(i, h)} \tag{3.52}$$

$$\boldsymbol{\mu}_{ig} = \frac{\sum_{t \in T(\boldsymbol{O},g)} \gamma'_t(i, g)\, V(\boldsymbol{o}_t)}{\sum_{t \in T(\boldsymbol{O},g)} \gamma'_t(i, g)}, \quad n_g > 0 \tag{3.53}$$

$$\boldsymbol{\Sigma}_{ig} = \frac{\sum_{t \in T(\boldsymbol{O},g)} \gamma'_t(i,g)(V(\boldsymbol{o}_t) - \boldsymbol{\mu}_{ig})(V(\boldsymbol{o}_t) - \boldsymbol{\mu}_{ig})^T}{\sum_{t \in T(\boldsymbol{O},g)} \gamma'_t(i,g)}, \quad n_g > 0 \tag{3.54}$$

where $\gamma_t(i,h)$ and $\xi_t(i,j)$ can be calculated by using the forward variable $\alpha_t(i)$ and backward variable $\beta_t(i)$ as follows:

$$\begin{aligned} \gamma_t(i,h) &= P(q_t = i, l_t = h|\boldsymbol{O}, \lambda) \\ &= \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \cdot \frac{w_{ih}\mathcal{N}_{ih}(V(\boldsymbol{o}_t))}{\sum_{g \in S(\boldsymbol{o}_t)} w_{ig}\mathcal{N}_{ig}(V(\boldsymbol{o}_t))} \end{aligned} \tag{3.55}$$

$$\begin{aligned} \xi_t(i,j) &= P(q_t = i, q_{t+1} = j|\boldsymbol{O}, \lambda) \\ &= \frac{\alpha_t(i)a_{ij}b_j(\boldsymbol{o}_{t+1})\beta_{t+1}(j)}{\sum_{h=1}^{N}\sum_{k=1}^{N} \alpha_t(h)a_{hk}b_k(\boldsymbol{o}_{t+1})\beta_{t+1}(k)} \end{aligned} \tag{3.56}$$

From the condition mentioned in Theorem 2, it can be shown that each $\boldsymbol{\Sigma}_{ig}$ is positive definite.

3.2.4 Application to F0 pattern modeling

The MSD-HMM includes the discrete HMM and the continuous mixture HMM as special cases since the multi-space probability distribution includes the discrete distribution and the continuous distribution. If $n_g \equiv 0$, the MSD-HMM is the same as the discrete HMM. In the case where $S(\boldsymbol{o}_t)$ specifies one space, i.e., $|S(\boldsymbol{o}_t)| \equiv 1$, the MSD-HMM is exactly the same as the conventional discrete HMM. If $|S(\boldsymbol{o}_t)| \geq 1$, the MSD-HMM is the same as the discrete HMM based on the multi-labeling VQ [26]. If $n_g \equiv m > 0$ and $S(\boldsymbol{o}) \equiv \{1, 2, \ldots, G\}$, the MSD-HMM is the same as the continuous G-mixture HMM. These can also be confirmed by the fact that if $n_g \equiv 0$ and $|S(\boldsymbol{o}_t)| \equiv 1$, the reestimation formulas (3.50)-(3.52) are the same as those for discrete HMM of codebook size G, and if $n_g \equiv m$ and $S(\boldsymbol{o}_t) \equiv \{1, 2, \ldots, G\}$, the reestimation formulas (3.50)-(3.54) are the same as those for continuous HMM with m-dimensional G-mixture densities. Further, the MSD-HMM can model the sequence of observation vectors with variable dimension including zero-dimensional observations, i.e., discrete symbols.

While the observation of F0 has a continuous value in the voiced region, there exist no value for the unvoiced region. We can model this kind of observation sequence assuming that the observed F0 value occurs from one-dimensional spaces and the

"unvoiced" symbol occurs from the zero-dimensional space defined in Section 3.2.1, that is, by setting $n_g = 1$ ($g = 1, 2, \ldots, G-1$), $n_G = 0$ and

$$S(\boldsymbol{o}_t) = \begin{cases} \{1, 2, \ldots, G-1\}, & \text{(voiced)} \\ \{G\}, & \text{(unvoiced)} \end{cases}, \tag{3.57}$$

the MSD-HMM can cope with F0 patterns including the unvoiced region without heuristic assumption. In this case, the observed F0 value is assumed to be drawn from a continuous $(G-1)$-mixture probability density function.

Chapter 4

Speech parameter generation from HMM

The performance of speech recognition based on HMMs has been improved by incorporating the dynamic features of speech. Thus we surmise that, if there is a method for speech parameter generation from HMMs which include the dynamic features, it will be useful for speech synthesis by rule. This chapter derives a speech parameter generation algorithm from HMMs which include the dynamic features.

4.1 Speech parameter generation based on maximum likelihood criterion

For a given continuous mixture HMM λ, we derive an algorithm for determining speech parameter vector sequence

$$\boldsymbol{O} = \left[\boldsymbol{o}_1^\top, \boldsymbol{o}_2^\top, \ldots, \boldsymbol{o}_T^\top\right]^\top \tag{4.1}$$

in such a way that

$$P(\boldsymbol{O}|\lambda) = \sum_{\text{all } \boldsymbol{Q}} P(\boldsymbol{O}, \boldsymbol{Q}|\lambda) \tag{4.2}$$

is maximized with respect to $\boldsymbol{O}$, where

$$\boldsymbol{Q} = \{(q_1, i_1), (q_2, i_2), \ldots, (q_T, i_T)\} \tag{4.3}$$

is the state and mixture sequence, i.e., (q, i) indicates the i-th mixture of state q. We assume that the speech parameter vector $\boldsymbol{o}_t$ consists of the static feature vector $\boldsymbol{c}_t = [c_t(1), c_t(2), \ldots, c_t(M)]^\top$ (e.g., cepstral coefficients) and dynamic feature

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

$$\Delta \boldsymbol{c}_t = \sum_{\tau=-L_-^{(1)}}^{L_+^{(1)}} w^{(1)}(\tau) \boldsymbol{c}_{t+\tau} \tag{4.4}$$

$$\Delta^2 \boldsymbol{c}_t = \sum_{\tau=-L_-^{(2)}}^{L_+^{(2)}} w^{(2)}(\tau) \boldsymbol{c}_{t+\tau}. \tag{4.5}$$

We have derived algorithms [11], [12] for solving the following problems:

Case 1. For given λ and $\boldsymbol{Q}$, maximize $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$ with respect to $\boldsymbol{O}$ under the conditions (4.4), (4.5).

Case 2. For a given λ, maximize $P(\boldsymbol{O}, \boldsymbol{Q}|\lambda)$ with respect to $\boldsymbol{Q}$ and $\boldsymbol{O}$ under the conditions (4.4), (4.5).

In this section, we will review the above algorithms and derive an algorithm for the problem:

Case 3. For a given λ, maximize $P(\boldsymbol{O}|\lambda)$ with respect to $\boldsymbol{O}$ under the conditions (4.4), (4.5).

4.1.1 Case 1: Maximizing $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$ with respect to $\boldsymbol{O}$

First, consider maximizing $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$ with respect to $\boldsymbol{O}$ for a fixed state and mixture sequence $\boldsymbol{Q}$. The logarithm of $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$ can be written as

$$\log P(\boldsymbol{O}|\boldsymbol{Q}, \lambda) = -\frac{1}{2} \boldsymbol{O}^\top \boldsymbol{U}^{-1} \boldsymbol{O} + \boldsymbol{O}^\top \boldsymbol{U}^{-1} \boldsymbol{M} + K \tag{4.6}$$

where

$$\boldsymbol{U}^{-1} = \mathrm{diag}\left[\boldsymbol{U}_{q_1,i_1}^{-1}, \boldsymbol{U}_{q_2,i_2}^{-1}, ..., \boldsymbol{U}_{q_T,i_T}^{-1}\right] \tag{4.7}$$

$$\boldsymbol{M} = \left[\boldsymbol{\mu}_{q_1,i_1}^\top, \boldsymbol{\mu}_{q_2,i_2}^\top, ..., \boldsymbol{\mu}_{q_T,i_T}^\top\right]^\top \tag{4.8}$$

$\boldsymbol{\mu}_{q_t,i_t}$ and $\boldsymbol{U}_{q_t,i_t}$ are the $3M \times 1$ mean vector and the $3M \times 3M$ covariance matrix, respectively, associated with i_t-th mixture of state q_t, and the constant K is independent of $\boldsymbol{O}$.

It is obvious that $P(\boldsymbol{O}|\boldsymbol{Q},\lambda)$ is maximized when $\boldsymbol{O}=\boldsymbol{M}$ without the conditions (4.4), (4.5), that is, the speech parameter vector sequence becomes a sequence of the mean vectors. Conditions (4.4), (4.5) can be arranged in a matrix form:

$$\boldsymbol{O}=\boldsymbol{W}\boldsymbol{C} \tag{4.9}$$

where

$$\begin{aligned}
\boldsymbol{C} &= [\boldsymbol{c}_1,\boldsymbol{c}_2,\ldots,\boldsymbol{c}_T]^\top && (4.10)\\
\boldsymbol{W} &= [\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_T]^\top && (4.11)\\
\boldsymbol{w}_t &= \left[\boldsymbol{w}_t^{(0)},\boldsymbol{w}_t^{(1)},\boldsymbol{w}_t^{(2)}\right] && (4.12)\\
\boldsymbol{w}_t^{(n)} &= [\underbrace{\boldsymbol{0}_{M\times M},\ldots,\boldsymbol{0}_{M\times M}}_{1\text{st}},w^{(n)}(-L_-^{(n)})\boldsymbol{I}_{M\times M},\\
&\quad \ldots,\underbrace{w^{(n)}(0)\boldsymbol{I}_{M\times M}}_{t\text{-th}},\ldots,\underbrace{w^{(n)}(L_+^{(n)})\boldsymbol{I}_{M\times M}}_{(t+L_+^{(n)})\text{-th}},\\
&\quad \underbrace{\boldsymbol{0}_{M\times M},\ldots,\boldsymbol{0}_{M\times M}}_{T\text{-th}}]^\top,\quad n=0,1,2 && (4.13)
\end{aligned}$$

$L_-^{(0)}=L_+^{(0)}=0$, and $w^{(0)}(0)=1$. Under the condition (4.9), maximizing $P(\boldsymbol{O}|\boldsymbol{Q},\lambda)$ with respect to $\boldsymbol{O}$ is equivalent to that with respect to $\boldsymbol{C}$. By setting

$$\frac{\partial \log P(\boldsymbol{W}\boldsymbol{C}|\boldsymbol{Q},\lambda)}{\partial \boldsymbol{C}}=\boldsymbol{0}, \tag{4.14}$$

we obtain a set of equations

$$\boldsymbol{W}^\top\boldsymbol{U}^{-1}\boldsymbol{W}\boldsymbol{C}=\boldsymbol{W}^\top\boldsymbol{U}^{-1}\boldsymbol{M}^\top. \tag{4.15}$$

For direct solution of (4.15), we need $O(T^3M^3)$ operations[1]because $\boldsymbol{W}^\top\boldsymbol{U}^{-1}\boldsymbol{W}$ is a $TM\times TM$ matrix. By utilizing the special structure of $\boldsymbol{W}^\top\boldsymbol{U}^{-1}\boldsymbol{W}$, (4.15) can be solved by the Cholesky decomposition or the QR decomposition with $O(TM^3L^2)$ operations[2], where $L=\max_{n\in\{1,2\},s\in\{-,+\}}L_s^{(n)}$. Equation (4.15) can also be solved by an algorithm derived in [11], [12], which can operate in a time-recursive manner [28].

4.1.2 Case 2: Maximizing $P(\boldsymbol{O},\boldsymbol{Q}|\lambda)$ with respect to $\boldsymbol{O}$ and $\boldsymbol{Q}$

This problem can be solved by evaluating $\max_{\boldsymbol{C}}P(\boldsymbol{O},\boldsymbol{Q}|\lambda)=\max_{\boldsymbol{C}}P(\boldsymbol{O}|\boldsymbol{Q},\lambda)P(\boldsymbol{Q}|\lambda)$ for all $\boldsymbol{Q}$. However, it is impractical because there are too many combinations of

[1]When $\boldsymbol{U}_{q,i}$ is diagonal, it is reduced to $O(T^3M)$ since each of the M-dimensions can be calculated independently.

[2]When $\boldsymbol{U}_{q,i}$ is diagonal, it is reduced to $O(TML^2)$. Furthermore, when $L_-^{(1)}=-1$, $L_+^{(1)}=0$, and $w^{(2)}(i)\equiv 0$, it is reduced to $O(TM)$ as described in [27].

$\boldsymbol{Q}$. We have developed a fast algorithm for searching for the optimal or sub-optimal state sequence keeping $\boldsymbol{C}$ optimal in the sense that $P(\boldsymbol{O}|\boldsymbol{Q},\lambda)$ is maximized with respect to $\boldsymbol{C}$ [11], [12].

To control temporal structure of speech parameter sequence appropriately, HMMs should incorporate state duration densities. The probability $P(\boldsymbol{O},\boldsymbol{Q}|\lambda)$ can be written as $P(\boldsymbol{O},\boldsymbol{Q}|\lambda) = P(\boldsymbol{O},\boldsymbol{i}|\boldsymbol{q},\lambda)P(\boldsymbol{q}|\lambda)$, where $\boldsymbol{q} = \{q_1, q_2, \ldots, q_T\}$, $\boldsymbol{i} = \{i_1, i_2, \ldots, i_T\}$, and the state duration probability $P(\boldsymbol{q}|\lambda)$ is given by

$$\log P(\boldsymbol{q}|\lambda) = \sum_{n=1}^{N} \log p_{q_n}(d_{q_n}) \tag{4.16}$$

where the total number of states which have been visited during T frames is N, and $p_{q_n}(d_{q_n})$ is the probability of d_{q_n} consecutive observations in state q_n. If we determine the state sequence $\boldsymbol{q}$ only by $P(\boldsymbol{q}|\lambda)$ independently of $\boldsymbol{O}$, maximizing $P(\boldsymbol{O},\boldsymbol{Q}|\lambda) = P(\boldsymbol{O},\boldsymbol{i}|\boldsymbol{q},\lambda)P(\boldsymbol{q}|\lambda)$ with respect to $\boldsymbol{O}$ and $\boldsymbol{Q}$ is equivalent to maximizing $P(\boldsymbol{O},\boldsymbol{i}|\boldsymbol{q},\lambda)$ with respect to $\boldsymbol{O}$ and $\boldsymbol{i}$. Furthermore, if we assume that state output probabilities are single-Gaussian, $\boldsymbol{i}$ is unique. Therefore, the solution is obtained by solving (4.15) in the same way as the Case 1.

4.1.3 Case 3: Maximizing $P(\boldsymbol{O}|\lambda)$ with respect to $\boldsymbol{O}$

We derive an algorithm based on an EM algorithm, which find a critical point of the likelihood function $P(\boldsymbol{O}|\lambda)$. An auxiliary function of current parameter vector sequence $\boldsymbol{O}$ and new parameter vector sequence $\boldsymbol{O}'$ is defined by

$$Q(\boldsymbol{O},\boldsymbol{O}') = \sum_{\text{all } \boldsymbol{Q}} P(\boldsymbol{O},\boldsymbol{Q}|\lambda) \log P(\boldsymbol{O}',\boldsymbol{Q}|\lambda). \tag{4.17}$$

It can be shown that by substituting $\boldsymbol{O}'$ which maximizes $Q(\boldsymbol{O},\boldsymbol{O}')$ for $\boldsymbol{O}$, the likelihood increases unless $\boldsymbol{O}$ is a critical point of the likelihood. Equation (4.17) can be written as

$$Q(\boldsymbol{O},\boldsymbol{O}') = P(\boldsymbol{O}|\lambda)\left\{-\frac{1}{2}\boldsymbol{O}'^{\top}\overline{\boldsymbol{U}^{-1}}\boldsymbol{O}' + \boldsymbol{O}'^{\top}\overline{\boldsymbol{U}^{-1}\boldsymbol{M}} + \overline{K}\right\} \tag{4.18}$$

where

$$\overline{\boldsymbol{U}^{-1}} = \operatorname{diag}\left[\overline{\boldsymbol{U}_1^{-1}}, \overline{\boldsymbol{U}_2^{-1}}, \ldots, \overline{\boldsymbol{U}_T^{-1}}\right] \tag{4.19}$$

$$\overline{\boldsymbol{U}_t^{-1}} = \sum_{q,i} \gamma_t(q,i)\boldsymbol{U}_{q,i}^{-1} \tag{4.20}$$

$$\overline{\boldsymbol{U}^{-1}\boldsymbol{M}} = \left[\overline{\boldsymbol{U}_1^{-1}\boldsymbol{\mu}_1}^{\top}, \overline{\boldsymbol{U}_2^{-1}\boldsymbol{\mu}_2}^{\top}, \ldots, \overline{\boldsymbol{U}_T^{-1}\boldsymbol{\mu}_T}^{\top}\right]^{\top} \tag{4.21}$$

$$\overline{\boldsymbol{U}_t^{-1}\boldsymbol{\mu}_t} = \sum_{q,i} \gamma_t(q,i)\,\boldsymbol{U}_{q,i}^{-1}\boldsymbol{\mu}_{q,i} \tag{4.22}$$

and the constant $\overline{K}$ is independent of $\boldsymbol{O}'$. The occupancy probability $\gamma_t(q,i)$ defined by

$$\gamma_t(q,i) = P(q_t = (q,i)|\boldsymbol{O},\lambda) \tag{4.23}$$

can be calculated with the forward-backward inductive procedure. Under the condition $\boldsymbol{O}' = \boldsymbol{W}\boldsymbol{C}'$, $\boldsymbol{C}'$ which maximizes $Q(\boldsymbol{O},\boldsymbol{O}')$ is given by the following set of equations:

$$\boldsymbol{W}^\top\overline{\boldsymbol{U}^{-1}}\boldsymbol{W}\boldsymbol{C}' = \boldsymbol{W}^\top\overline{\boldsymbol{U}^{-1}\boldsymbol{M}}. \tag{4.24}$$

The above set of equations has the same form as (4.15). Accordingly, it can be solved by the algorithm for solving (4.15).

The whole procedure is summarized as follows:

Step 0. Choose an initial parameter vector sequence $\boldsymbol{C}$.

Step 1. Calculate $\gamma_t(q,i)$ with the forward-backward algorithm.

Step 2. Calculate $\overline{\boldsymbol{U}^{-1}}$ and $\overline{\boldsymbol{U}^{-1}\boldsymbol{M}}$ by (4.19)–(4.22), and solve (4.24).

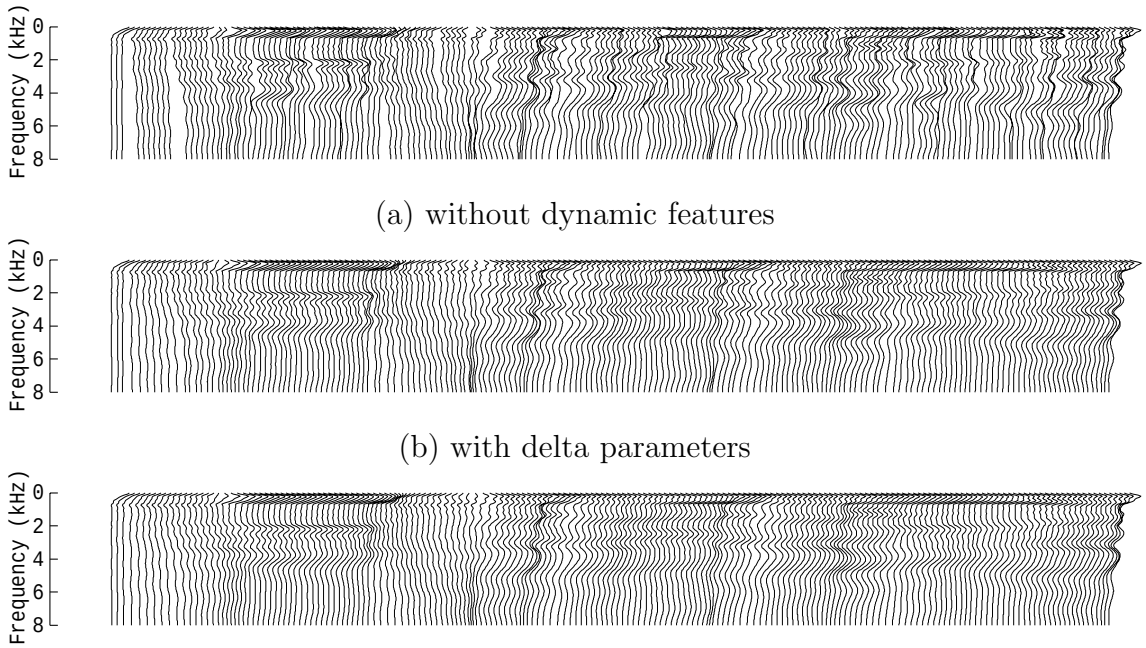
Step 3. Set $\boldsymbol{C} = \boldsymbol{C}'$. If a certain convergence condition is satisfied, stop; otherwise, goto Step 1.

From the same reason as Case 2, HMMs should incorporate state duration densities. If we determine the state sequence $\boldsymbol{q}$ only by $P(\boldsymbol{q}|\lambda)$ independently of $\boldsymbol{O}$ in a manner similar to the previous section, only the mixture sequence $\boldsymbol{i}$ is assumed to be unobservable[3]. Further, we can also assume that $\boldsymbol{Q}$ is unobservable but phoneme or syllable durations are given.

4.2 Example

A simple experiment of speech synthesis was carried out using the parameter generation algorithm. We used phonetically balanced 450 sentences from ATR Japanese speech database for training. The type of HMM used was a continuous Gaussian model. The diagonal covariances were used. All models were 5-state left-to-right models with no skips. The heuristic duration densities were calculated after the training. Feature vector consists of 25 mel-cepstral coefficients including the zeroth coefficient, their delta and delta-delta coefficients. Mel-cepstral coefficients were obtained by the mel-cepstral analysis. The signal was windowed by a 25ms Black man window with a 5ms shift.

[3]For this problem, an algorithm based on a direct search has also been proposed in [29].



(a) without dynamic features

(b) with delta parameters

(c) with delta and delta-delta parameters

Figure 4.1: Spectra generated with dynamic features for a Japanese phrase "chi-isanaunagi".

4.2.1 Effect of dynamic feature

First, we observed the parameter generation in the case 1, in which parameter sequence O maximizes $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$. State sequence Q was estimated from the result of Veterbi alignment of natural speech. Fig. 4.1 shows the spectra calculated from the mel-cepstral coefficients generated by the HMM, which is composed by concatenation of phoneme models. Without the dynamic features, the parameter sequence which maximizes $P(\boldsymbol{O}|\boldsymbol{Q}, \lambda)$ becomes a sequence of the mean vectors (Fig. 4.1(a)). On the other hand, Fig. 4.1(b) and Fig. 4.1(c) show that appropriate parameter sequences are generated by using the static and dynamic feature. Looking at Fig. 4.1(b) and Fig. 4.1(c) closely, we can see that incorporation of delta-delta parameter improves smoothness of generated speech spectra.

Fig 4.2 shows probability density functions and generated parameters for the zero-th mel-cepstral coefficients. The x-axis represents frame number. A gray box and its middle line represent standard deviation and mean of probability density function, respectively, and a curved line is the generated zero-th mel-cepstral coefficients. From this figure, it can be observed that parameters are generated taking account of constraints of their probability density function and dynamic features.

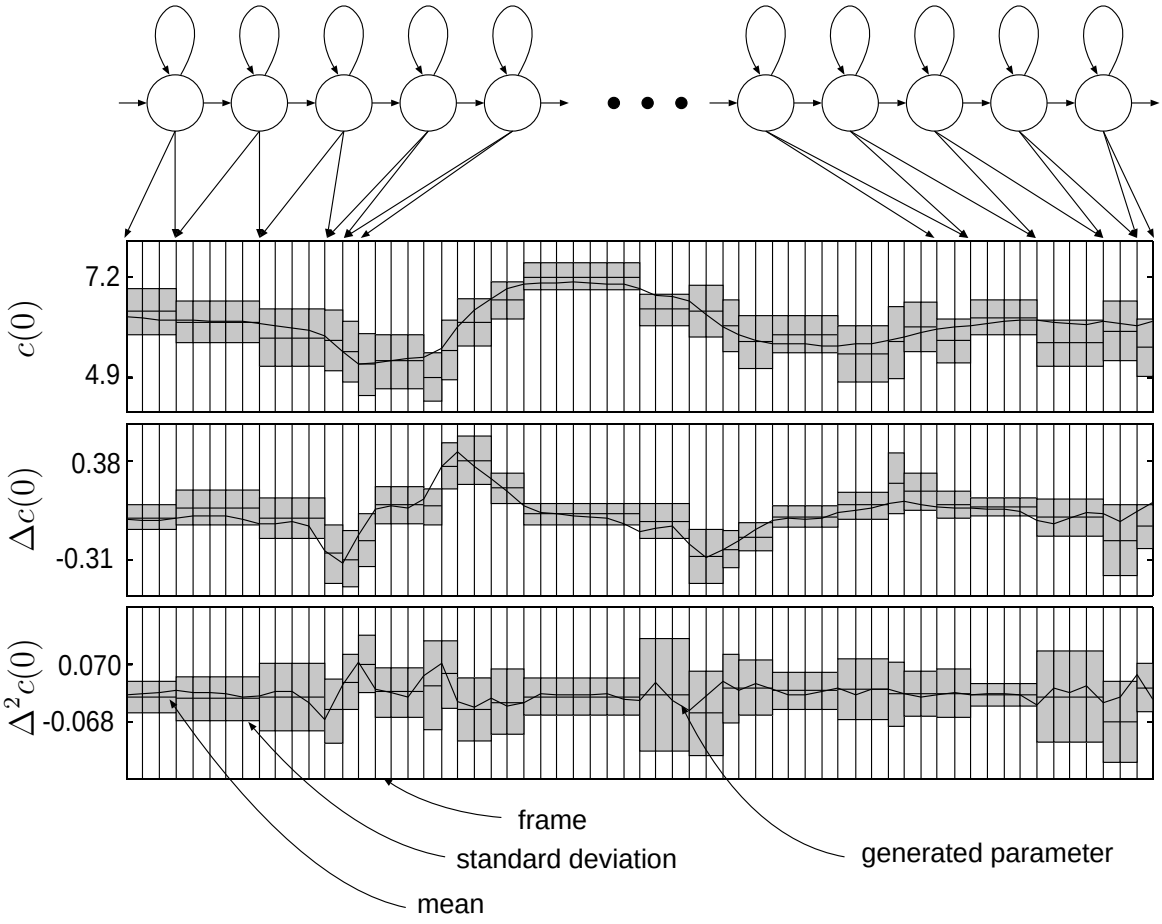


Figure 4.2: Relation between probability density function and generated parameter for a Japanese phrase "unagi" (top: static, middle: delta, bottom: delta-delta).

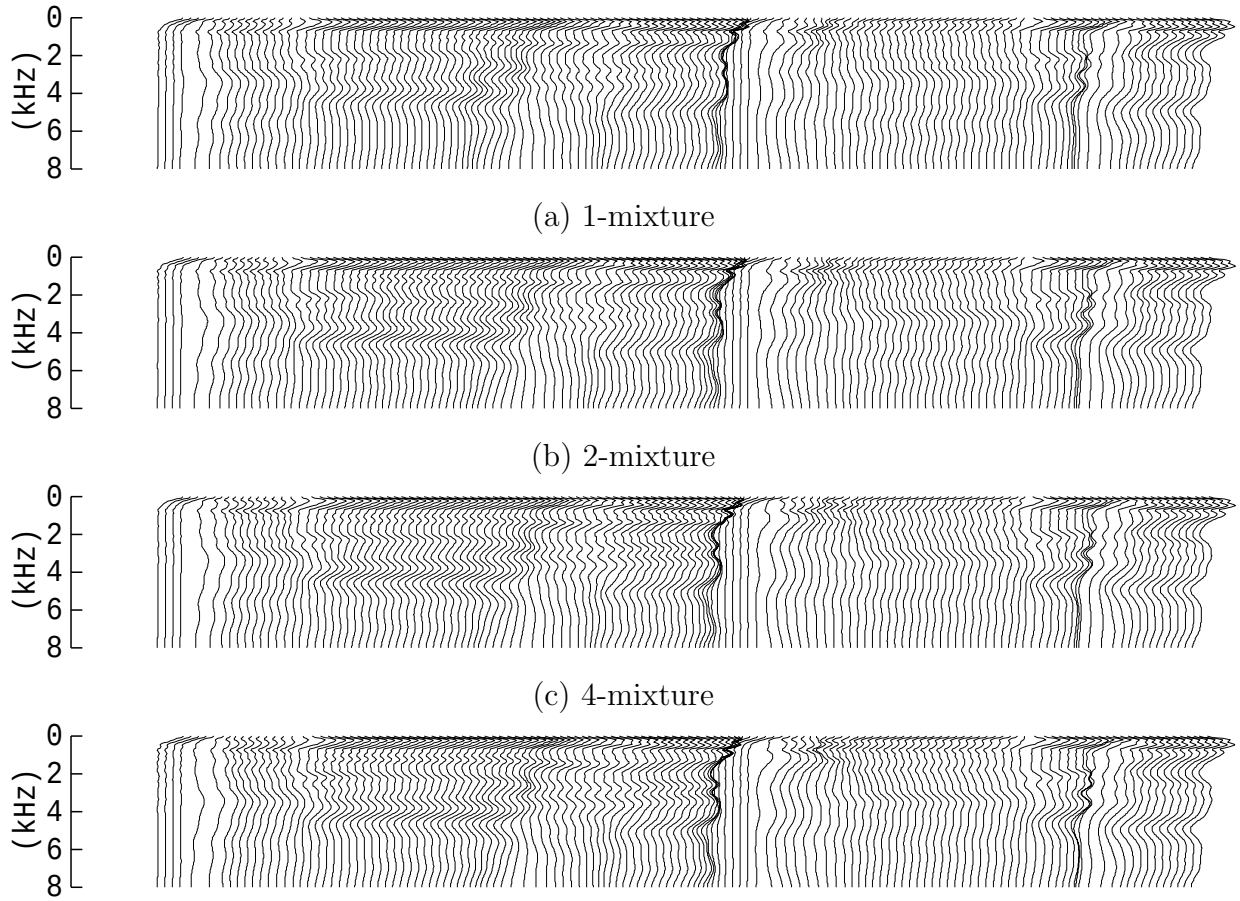
4.2.2 Parameter generation using multi-mixuture HMM

In the algorithm of the case 3, we assumed that the state sequence (state and mixture sequence for the multi-mixture case) or a part of the state sequence is unobservable (i.e., hidden or latent). As a result, the algorithm iterates the forward-backward algorithm and the parameter generation algorithm for the case where state sequence is given. Experimental results show that by using the algorithm, we can reproduce clear formant structure from multi-mixture HMMs as compared with that produced from single-mixture HMMs.

It has found that a few iterations are sufficient for convergence of the proposed algorithm. Fig. 4.3 shows generated spectra for a Japanese sentence fragment "kiN-zokuhiroo" taken from a sentence which is not included in the training data. Fig. 4.4 compares two spectra obtained from single-mixure HMMs and 8-mixture HMMs, respectively, for the same temporal position of the sentence fragment. It is seen from Fig. 4.3 and Fig. 4.4 that with increasing mixtures, the formant structure of the generated spectra get clearer.

We evaluated two synthesized speech obtained from single-mixure HMMs and 8-mixture HMMs by listening test, where fundamental frequency and duration is obtained from natural speech. Figure 4.5 shows the result of pair comparison test. From the listening test, it has been observed that the quality of the synthetic speech is considerably improved by increasing mixtures.

When we use single-mixture HMMs, the formant structure of spectrum corresponding to each mean vector $\boldsymbol{\mu}_{q,i}$ might be vague since $\boldsymbol{\mu}_{q,i}$ is the average of different speech spectra. One can increase the number of decision tree leaf clusters. However, it might result in perceivable discontinuities in synthetic speech since overly large tree will be overspecialized to training data and generalized poorly. We expect that the proposed algorithm can avoid this situation in a simple manner.



(a) 1-mixture

(b) 2-mixture

(c) 4-mixture

(d) 8-mixture

Figure 4.3: Generated spectra for a sentence fragment "kiNzokuhiroo."

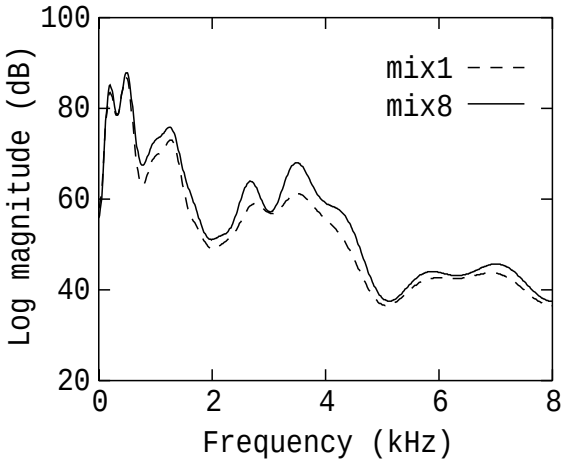


Figure 4.4: Spectra obtained from 1-mixture HMMs and 8-mixture HMMs.

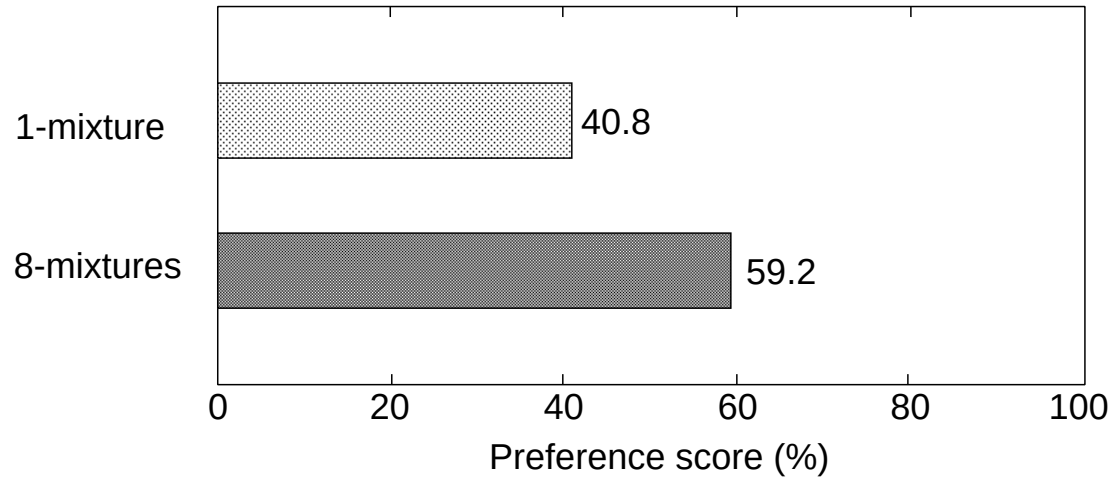


Figure 4.5: The result of the pair comparison test.

Chapter 5

Construction of HMM-based Text-to-Speech System

Phonetic parameter and prosodic parameter are modeled simultaneously in a unified framework of HMM. In the proposed system, mel-cepstrum, fundamental frequency (F0) and state duration are modeled by continuous density HMMs, multi-space probability distribution HMMs and multi-dimensional Gaussian distributions, respectively. The distributions for spectrum, F0, and the state duration are clustered independently by using a decision-tree based context clustering technique. This chapter describes feature vector modeled by HMM, structure of HMM and how to train context-dependent HMM.

5.1 Calculation of dynamic feature

In this thesis, mel-cepstral coefficient is used as spectral parameter. Mel-cepstral coefficient vectors $\boldsymbol{c}$ are obtained from speech database using a mel-cepstral analysis technique [16]. Their dynamic feature $\Delta\boldsymbol{c}$ and $\Delta^2\boldsymbol{c}$ are calculated as follows:

$$\Delta\boldsymbol{c}_t = -\frac{1}{2}\boldsymbol{c}_{t-1} + \frac{1}{2}\boldsymbol{c}_{t+1}, \tag{5.1}$$

$$\Delta^2\boldsymbol{c}_t = \frac{1}{4}\boldsymbol{c}_{t-1} - \frac{1}{2}\boldsymbol{c}_t + \frac{1}{4}\boldsymbol{c}_{t+1}. \tag{5.2}$$

In the same way, dynamic features for F0 are calculated by

$$\delta p_t = -\frac{1}{2}p_{t-1} + \frac{1}{2}p_{t+1}, \tag{5.3}$$

$$\delta^2 p_t = \frac{1}{4}p_{t-1} - \frac{1}{2}p_t + \frac{1}{4}p_{t+1}. \tag{5.4}$$

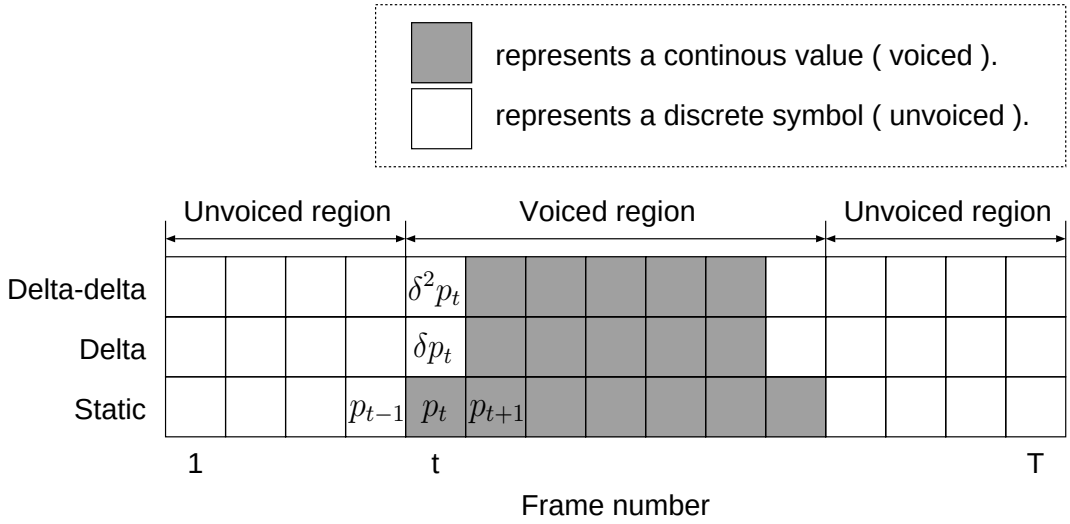


Figure 5.1: Calculation of dynamic features for F0.

where, in unvoiced region, p_t, δp_t and $\delta^2 p_t$ are defined as a discrete symbol. When dynamic features at the boundary between voiced and unvoiced can not be calculated, they are defined as a discrete symbol. For example, if dynamic features are calculated by Eq.(5.3)(5.4), δp_t and δ_t^p at the boundary between voiced and unvoiced as shown Fig. 5.1 become discrete symbol.

5.2 Spectrum and F0 modeling

In the chapter 3, it is described that sequence of mel-cepstral coefficient vector and F0 pattern are modeled by a continuous density HMM and multi-space probability distribution HMM, respectively.

We construct spectrum and F0 models by using embedded training because the embedded training does not need label boundaries when appropriate initial models are available. However, if spectrum models and F0 models are embedded-trained separately, speech segmentations may be discrepant between them.

To avoid this problem, context dependent HMMs are trained with feature vector which consists of spectrum, F0 and their dynamic features (Fig. 5.2). As a result, HMM has four streams as shown in Fig. 5.3.

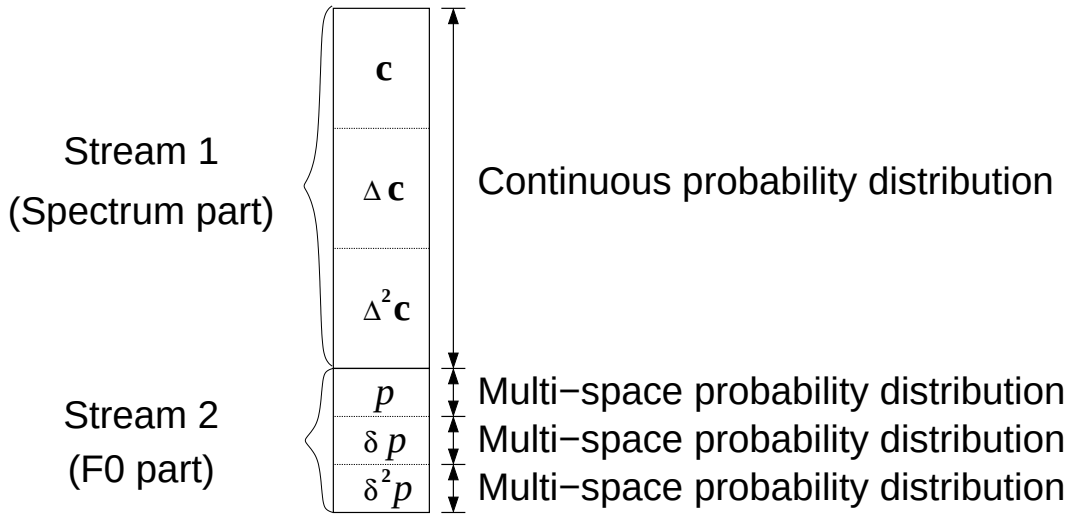


Figure 5.2: Feature vector.

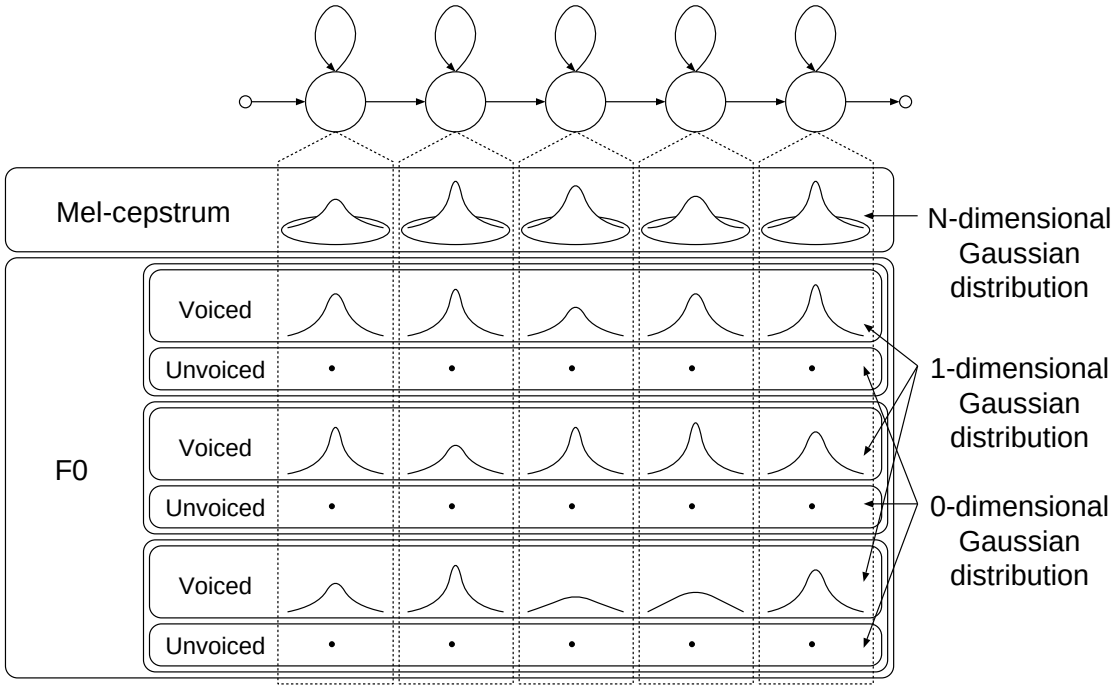


Figure 5.3: structure of HMM.

5.3 Duration modeling

5.3.1 Overview

There have been proposed techniques for training HMMs and their state duration densities simultaneously (e.g., [30]). However, these techniques require a large storage and computational load. In this thesis, state duration densities are estimated by using state occupancy probabilities which are obtained in the last iteration of embedded re-estimation [31].

In the HMM-based speech synthesis system described above, state duration densities were modeled by single Gaussian distributions estimated from histograms of state durations which were obtained by the Viterbi segmentation of training data. In this procedure, however, it is impossible to obtain variances of distributions for phonemes which appear only once in the training data.

In this thesis, to overcome this problem, Gaussian distributions of state durations are calculated on the trellis(Section 3.1.2) which is made in the embedded training stage. State durations of each phoneme HMM are regarded as a multi-dimensional observation, and the set of state durations of each phoneme HMM is modeled by a multi-dimensional Gaussian distribution. Dimension of state duration densities is equal to number of state of HMMs, and nth dimension of state duration densities is corresponding to nth state of HMMs[1]. Since state durations are modeled by continuous distributions, our approach has the following advantages:

- The speaking rate of synthetic speech can be varied easily.
- There is no need for label boundaries when appropriate initial models are available since the state duration densities are estimated in the embedded training stage of phoneme HMMs.

In the following sections, we describe training and clustering of state duration models, and determination of state duration in the synthesis part.

5.3.2 Training of state duration models

There have been proposed techniques for training HMMs and their state duration densities simultaneously, however, these techniques is inefficient because it requires huge storage and computational load. From this point of view, we adopt another technique for training state duration models.

[1]We assume the left-to-right model with no skip.

State duration densities are estimated on the trellis which is obtained in the embedded training stage. The mean $\xi(i)$ and the variance $\sigma^2(i)$ of duration density of state i are determined by

$$\xi(i) = \frac{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)(t_1-t_0+1)}{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)}, \tag{5.5}$$

$$\sigma^2(i) = \frac{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)(t_1-t_0+1)^2}{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(i)} - \xi^2(i), \tag{5.6}$$

respectively, where $\chi_{t_0,t_1}(i)$ is the probability of occupying state i from time t_0 to t_1 and can be written as

$$\chi_{t_0,t_1}(i) = (1-\gamma_{t_0-1}(i)) \cdot \prod_{t=t_0}^{t_1}\gamma_t(i) \cdot (1-\gamma_{t_1+1}(i)), \tag{5.7}$$

where $\gamma_t(i)$ is the occupation probability of state i at time t, and we define $\gamma_{-1}(i) = \gamma_{T+1}(i) = 0$.

5.4 Context dependent model

5.4.1 Contextual factors

There are many contextual factors (e.g., phone identity factors, stress-related factors, locational factors) that affect spectrum, F0 and duration. In this thesis, following contextual factors are taken into account:

- mora[2] count of sentence
- position of breath group in sentence
- mora count of {preceding, current, succeeding} breath group
- position of current accentual phrase in current breath group
- mora count and accent type of {preceding, current, succeeding} accentual phrase

[2]A mora is a syllable-sized unit in Japanese.

- {preceding, current, succeeding} part-of-speech
- position of current phoneme in current accentual phrase
- {preceding, current, succeeding} phoneme

Note that a context dependent HMM corresponds to a phoneme.

5.4.2 Decision-tree based context clustering

When we construct context dependent models taking account of many combinations of the above contextual factors, we expect to be able to obtain appropriate models. However, as contextual factors increase, their combinations also increase exponentially. Therefore, model parameters with sufficient accuracy cannot be estimated with limited training data. Furthermore, it is impossible to prepare speech database which includes all combinations of contextual factors.

Introduction of context clustering

To overcome the above problem, we apply a decision-tree based context clustering technique [32] to distributions for spectrum, F0 and state duration.

The decision-tree based context clustering algorithm have been extended for MSD-HMMs in [33]. Since each of spectrum, F0 and duration have its own influential contextual factors, the distributions for spectral parameter and F0 parameter and the state duration are clustered independently (Fig. 5.4.2).

Example of decision tree

We used phonetically balanced 450 sentences from ATR Japanese speech database for training. Speech signals were sampled at 16 kHz and windowed by a 25-ms Blackman window with a 5-ms shift, and then mel-cepstral coefficients were obtained by the mel-cepstral analysis[3]. Feature vector consists of spectral and F0 parameter vectors. Spectral parameter vector consists of 25 mel-cepstral coefficients including the zeroth coefficient, their delta and delta-delta coefficients. F0 parameter vector consists of log F0, its delta and delta-delta. We used 3-state left-to-right HMMs with single diagonal Gaussian output distributions. Decision trees for spectrum, F0 and duration models were constructed as shown in Fig. 5.4.2. The resultant trees

[3]The source codes of the mel-cepstral analysis/synthesis can be found in *http://kt−lab.ics.nitech.ac.jp/˜tokuda/SPTK/* .

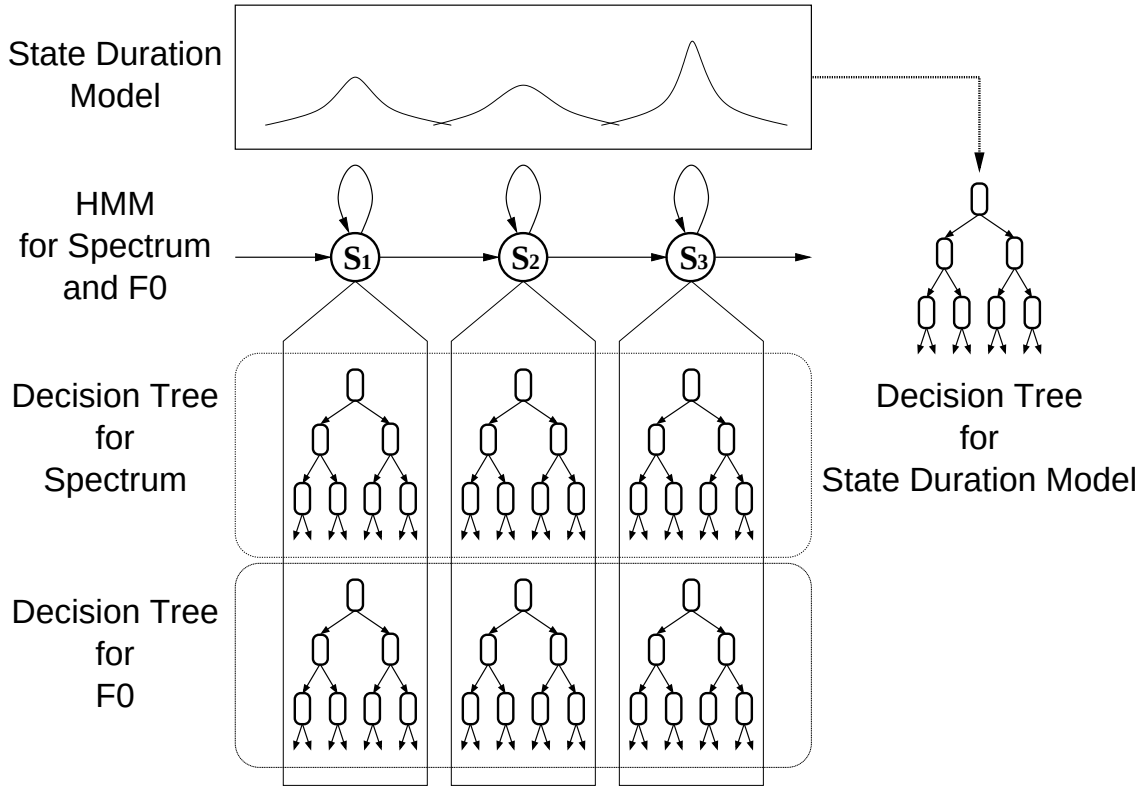
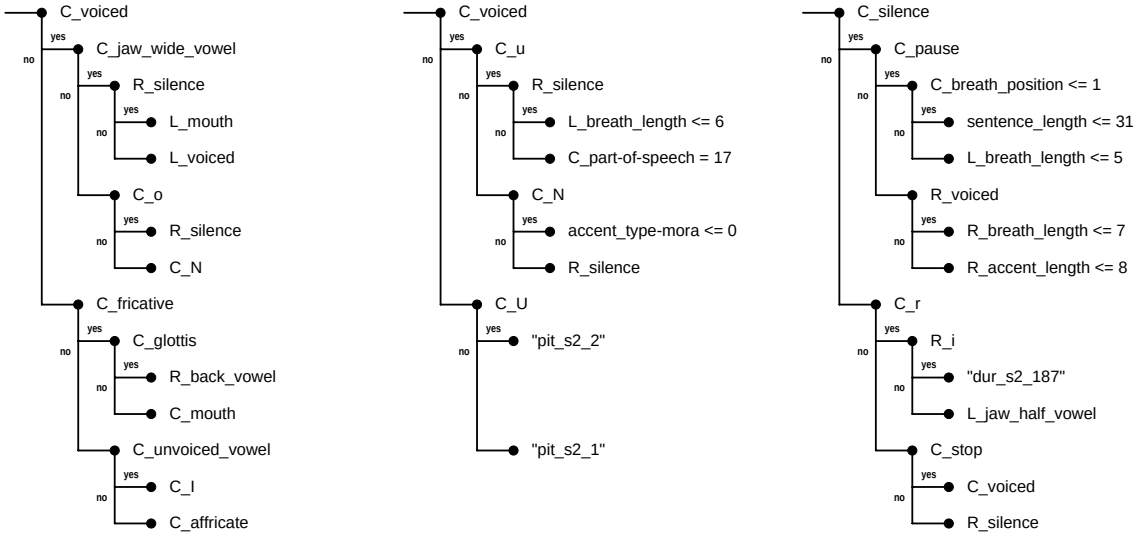


Figure 5.4: Decision trees.



(a) Tree for Spectrum Model(1st state)

(b) Tree for F0 Model(1st state)

(c) Tree for State Duration Model

Figure 5.5: Examples of decision trees.

for spectrum models, F0 models and state duration models had 6,615, 1,877 and 201 leaves in total, respectively.

Fig. 5.4.2 shows examples of constructed decision trees for spectrum (a), F0 (b) and state duration (c). In these figures, "L_∗", "C_∗" and "R_∗" represent "preceding", "current" and "succeeding", respectively. "Silence" represents silence of head or tail of a sentence, or pause. Questions of breath group and accentual phrase are represented by "∗_breath_∗" and "∗_accent_∗", respectively. "Pit_s2_∗" and "dur_s2_∗" represent leaf nodes. From these figures, it is seen that spectrum models are much affected by phonetic identity, F0 models for "voiced" are much affected by accentual phrase and part-of-speech, and F0 models for "unvoiced" are clustered by a very simple tree. With regard to state duration models, it can be seen that silence and pause models are much affected by accentual phrase and part-of-speech, and the other models are much affected by phonetic identity.

Through informal listening tests, we have found that the stopping rule (a minimum frame occupancy at each leaf and a minimum gain in likelihood per splice) should be determined appropriately in decision tree construction. An overly large tree will be overspecialized to training data and generalize poorly. On the other hand, a overly small tree will model the data badly. Therefore we should introduce some stopping criterion or cross-validation method (e.g., [34]–[35]).

5.4.3 Context clustering using MDL principle

In this section, the MDL principle is incorporated as the stopping rule for the tree-based clustering [36], Before explaining the use of the MDL principle for the tree-based clustering, this section briefly introduces the principle itself.

Introduction of MDL principle

Minimum Description Length (MDL) principle is an information criterion which has been proven to be effective in the selection of optimal probabilistic models. It has been used to obtain a good model for data in various problems. According to the MDL principle, the model with the minimum description length l is selected to be the optimal model for data $x^N = x_1, \cdots, x_N$, from among probabilistic models $i = 1, \cdots, I$. A description length $l(i)$ of model i is given as,

$$l(i) = -\log P_{\hat{\theta}^{(i)}}(x^N) + \frac{\alpha_i}{2}\log N + \log I \tag{5.8}$$

where α_i is the number of free parameters of the model i, and $\hat{\theta}^{(i)}$ is the maximum likelihood estimates for the parameters $\theta(i) = (\theta_1^{(i)}, \cdots, \theta_{\alpha_i}^{(i)})$ of model i. The first term is the negative of the log likelihood for the data, and the second term represents the complexity of the model. The third term is the description length required for choosing model i. As a model becomes more complex, the value of the first term decreases and that of the second term increases. The description length l has its minimum at a model of appropriate complexity. Furthermore, as one can see in Eq. (5.8), the MDL principle does not need any externally given parameters; the optimal model for the data is automatically obtained once a set of models is given.

MDL principle for tree-based clustering

The MDL principle is incorporated to the tree-based clustering for MSD-HMM which includes continuous density HMM. It is assumed that cluster set $\mathcal{S}$ which is a result of clustering is defined by

$$\mathcal{S} = \{S_1, S_2, \cdots, S_i, \cdots, S_M\} \tag{5.9}$$

Then, log likelihood $\mathcal{L}$ is calculated as follows:

$$\begin{aligned}\mathcal{L} &= -\sum_{s\in\mathcal{S}}\sum_{g=1}^{G}\frac{1}{2}(n_g(\log(2\pi)+1) + \log|\boldsymbol{\Sigma}_{sg}| \\ &\qquad -2\log w_{sg})\sum_{t\in T(\boldsymbol{O},g)}\gamma_t(s,g),\end{aligned} \tag{5.10}$$

where g is a space index, and w_{sg} is a weight of space g in cluster s. $T(\boldsymbol{O}, g)$ is a set of time t which satisfies that a set of space index of observation vector $\boldsymbol{m}o_t$ includes a space index g. $\gamma_t(s, g)$ is the probability of being in space g of cluster s at time t. In the case of the zero-dimensional space, $\log|\boldsymbol{\Sigma}_{sg}|$ in Eq.(5.10) is equal to 0. Using this likelihood $\mathcal{L}$, a description length (DL) l is represented by

$$\begin{aligned} l &= \sum_{s\in\mathcal{S}}\sum_{g=1}^{G}\frac{1}{2}(n_g(\log(2\pi)+1)+\log|\boldsymbol{\Sigma}_{sg}| \\ &\qquad -2\log w_{sg})\sum_{t\in T(\boldsymbol{O},g)}\gamma_t(s,g) \\ &+ \left(\sum_{s\in\mathcal{S}}\sum_{g=1}^{G}\frac{1}{2}(2n_g+1)\right) \\ &\qquad \cdot\left(\log\sum_{s\in\mathcal{S}}\sum_{g=1}^{G}\sum_{t\in T(\boldsymbol{O},g)}\gamma_t(s,g)\right) \end{aligned} \tag{5.11}$$

If it is assumed that the description length is l' when a cluster S_i is divided two clusters S_{i+} and S_{i-}, The change of description length δl is calculated by

$$\begin{aligned} \delta l &= l' - l \\ &= \sum_{s\in\{S_{i+},S_{i-}\}}\sum_{g=1}^{G}\frac{1}{2}\left(\log|\boldsymbol{\Sigma}_{sg}|-2\log w_{sg}\right) \\ &\qquad \cdot\sum_{t\in T(\boldsymbol{O},g)}\gamma_t(s,g) \\ &- \sum_{s\in\{S_i\}}\sum_{g=1}^{G}\frac{1}{2}\left(\log|\boldsymbol{\Sigma}_{sg}|-2\log w_{sg}\right) \\ &\qquad \cdot\sum_{t\in T(\boldsymbol{O},g)}\gamma_t(s,g) \\ &+ \left(\sum_{g=1}^{G}\frac{1}{2}(2n_g+1)\right) \\ &\qquad \cdot\left(\log\sum_{s\in\mathcal{S}}\sum_{g=1}^{G}\sum_{t\in T(\boldsymbol{O},g)}\gamma_t(s,g)\right). \end{aligned} \tag{5.12}$$

If $\delta l < 0$ then the node is devided, and if $\delta l \geq 0$ then the node is not devided. Fig. 5.4.2 shows examples of constructed decision trees constructed by using the MDL principle.

Chapter 6

HMM-based Text-to-Speech Synthesis

Synthetic speech is generated by using an speech parameter generation algorithm from HMM and a mel-cepstrum based vocoding technique. Through informal listening tests, we have confirmed that the proposed system successfully synthesizes natural-sounding speech which resembles the speaker in the training database.

6.1 Overview

The synthesis part of the HMM-based text-to-speech synthesis system is shown in Fig. 6.1. In the synthesis part, an arbitrarily given text to be synthesized is converted to a context-based label sequence. Then, according to the label sequence, a sentence HMM is constructed by concatenating context dependent HMMs. State durations of the sentence HMM are determined so as to maximize the likelihood of the state duration densities [31]. According to the obtained state durations, a sequence of mel-cepstral coefficients and F0 values including voiced/unvoiced decisions is generated from the sentence HMM by using the speech parameter generation algorithm [12]. Finally, speech is synthesized directly from the generated mel-cepstral coefficients and F0 values by the MLSA filter [16], [19].

6.2 Text analysis

The inputed text is converted a context dependent label sequence by a text analyzer. For the TTS system, the text analyzer should have ability to extracted contextual

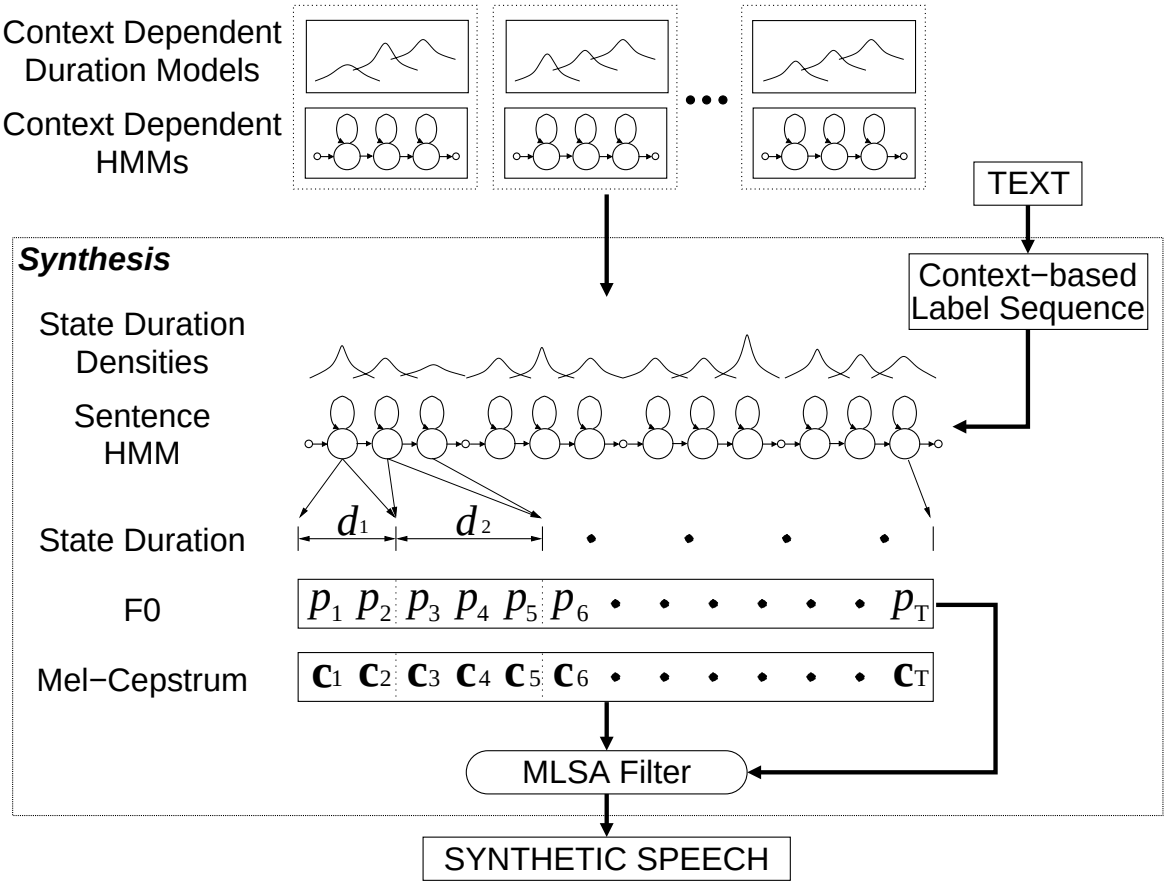


Figure 6.1: The block diagram of the HMM-based TTS.

informations listed in Section 5.4.1 from the text. Currently, some Japanese text analyzer (e.g., ChaSen[1], MSLR[2], etc.) are freely available. However, no text analyzer has the ability to extract accentual phrase and to decide accent type of accentual phrase. Thus, in this thesis, the inputed text is manually analyzed and converted context dependent label sequence.

6.3 Duration determination

For a given speech length T, the goal is to obtain a state sequence $\mathbf{q} = \{q_1, q_2, \cdots, q_T\}$ which maximize

$$\log P(\mathbf{q}|\lambda, T) = \sum_{k=1}^{K} \log p_k(d_k) \tag{6.1}$$

under the constraint

$$T = \sum_{k=1}^{K} d_k, \tag{6.2}$$

where $p_k(d_k)$ is the probability of duration d_k in state k, and K is the number of states in HMM λ.

Since each duration density $p_k(d_k)$ is modeled by a single Gaussian distribution, state durations $\{d_k\}_{k=1}^{K}$ which maximize (6.1) are given by

$$d_k = \xi(k) + \rho \cdot \sigma^2(k) \tag{6.3}$$

$$\rho = \left(T - \sum_{k=1}^{K} \xi(k)\right) \Big/ \sum_{k=1}^{K} \sigma^2(k) \ , \tag{6.4}$$

where $\xi(k)$ and $\sigma^2(k)$ are the mean and variance of the duration density of state k, respectively.

Since ρ is associated with T in (6.4), the speaking rate can be controlled by ρ instead of T. From (6.3), it can be seen that to synthesize speech with average speaking rate, ρ should be set to 0, that is, $T = \sum_{k=1}^{K} \xi(k)$, and the speaking rate becomes faster or slower when we set ρ to negative or positive value, respectively. It can also be seen that the variance $\sigma^2(k)$ represents "elasticity" of kth state duration.

6.4 Speech parameter generation

According to the estimated state duration, spectral and excitation parameters are generated from a sentence HMM constructed by concatenating context dependent

[1] *http://chasen.aist-nara.ac.jp/*
[2] *http://tanaka-www.cs.titech.ac.jp/pub/mslr/*

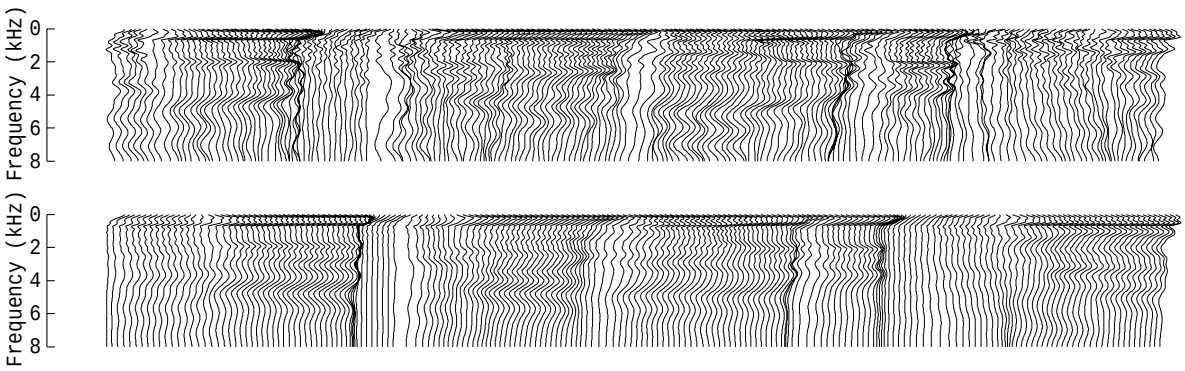


Figure 6.2: Generated spectra for a phrase "heikiNbairitsu" (top: natural spectra, bottom: generated spectra).

HMMs. Fig. 6.2 shows spectra of natural speech and synthesized speech for a Japanese phrase "heikiNbairitu" Fig. 6.4 shows F0 pattern of natural speech and synthesized speech for a Japanese sentence "heikiNbairituwo sageta keisekiga aru".

6.5 Experiments

6.5.1 Effect of dynamic feature

Examples of generated spectra and F0 pattern are shown in Fig. 6.5.1 and Fig. 6.5.1, respectively. In each figure, the parameters generated with and without dynamic features are shown. From these figures, it can be seen that spectra and F0 pattern which approximate those of natural speech are generated by using the parameter generation algorithm with dynamic features.

The effect of speech parameter generation with the dynamic features was evaluated by a pair comparison test. The following four samples were compared:

- spectrum generated without dynamic features
  + F0 without dynamic features
- spectrum generated with dynamic features
  + F0 without dynamic features
- spectrum generated without dynamic features

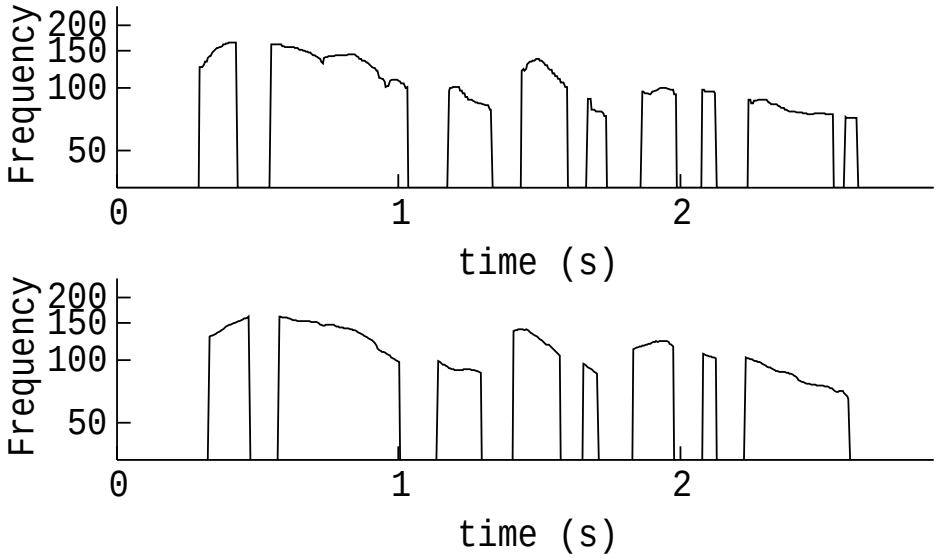
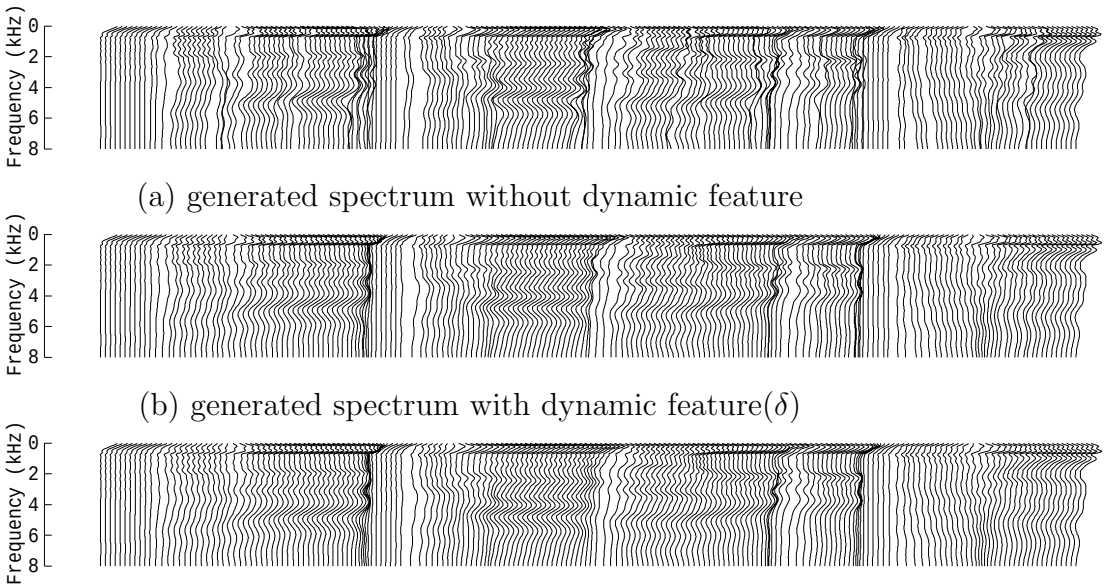


Figure 6.3: Generated F0 pattern for a sentence "heikiNbairitsuwo sageta keisekiga aru" (top: natural F0 pattern, bottom: generated F0 pattern).

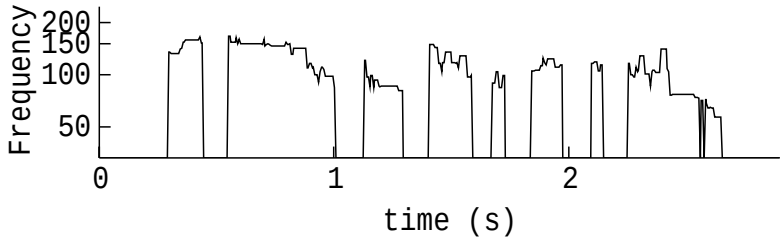


(a) generated spectrum without dynamic feature

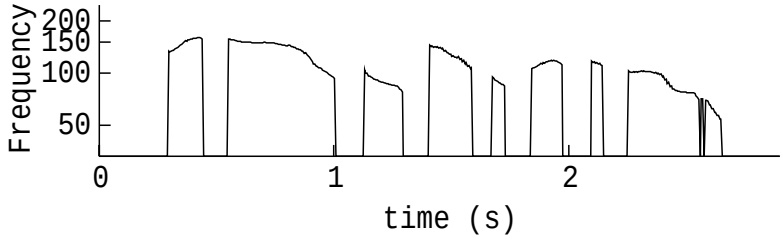
(b) generated spectrum with dynamic feature(δ)

(c) generated spectrum with dynamic feature($\delta + \delta^2$)

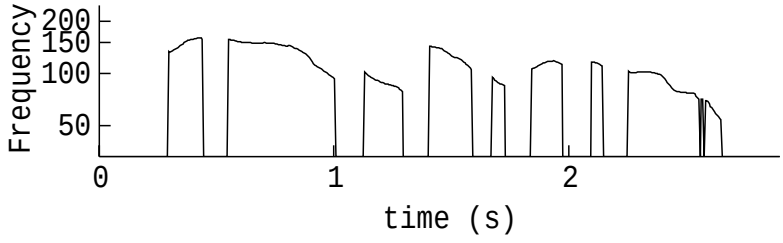
Figure 6.4: Generated spectra for a phrase "heikiNbairitsu."



(a) generated F0 without dynamic feature



(b) generated F0 with dynamic feature(δ)



(c) generated F0 with dynamic feature($\delta + \delta^2$)

Figure 6.5: Generated F0 pattern for a sentence "heikiNbairitsuwo sageta keisekiga aru."

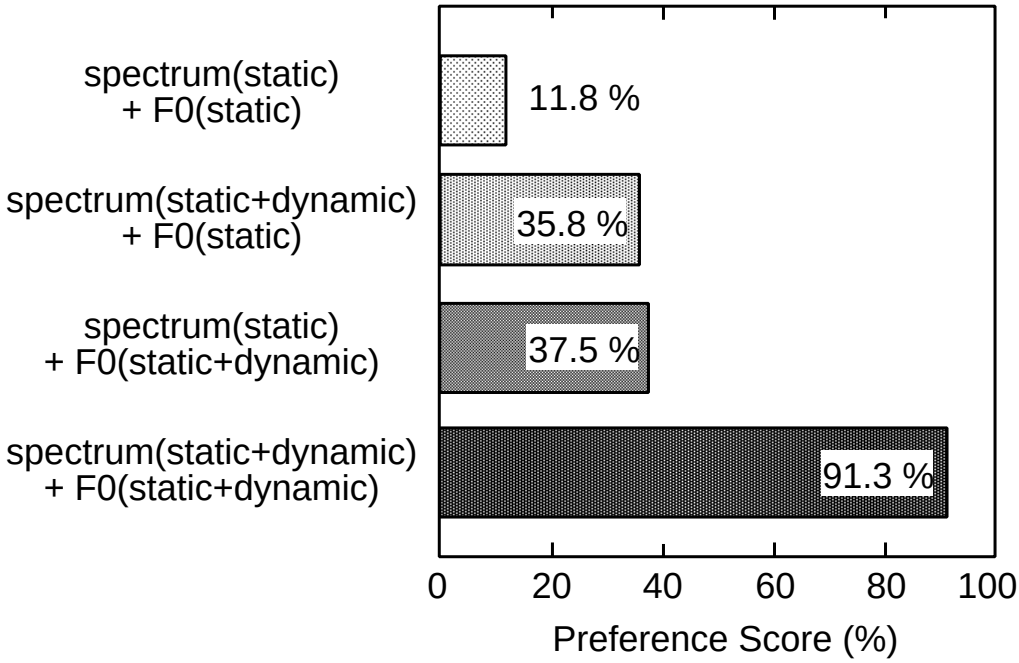


Figure 6.6: Effect of dynamic feature.

+ F0 with dynamic features

- spectrum generated with dynamic features
  + F0 with dynamic features

Figure 6.5.1 shows preference scores. It can be seen that in the case of using dynamic features synthesized speech is significantly improved.

6.5.2 Automatically system training

In the proposed system, there is no need for label boundaries when appropriate initial models are available since spectral, F0 and duration models are estimated by the the embedded training. Therefore, the system can be automatically constructed by the following process:

1. As a initial model, speaker independent and gender dependent model is prepared.

2. The target speaker dependent model is estimated by using initial model and speech data with transcription.

Table 6.1: Number of distribution.

	Spectrum	F0	Duration
SD	942	2597	1528
GD	952	2539	1552
SI	956	2623	1765

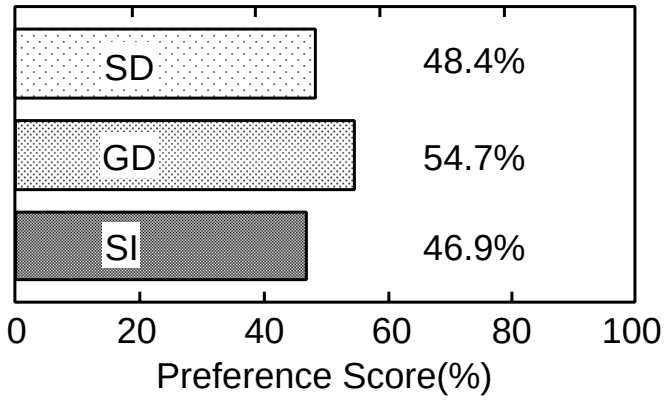


Figure 6.7: Effect of difference of initial model.

We did automatic system construction tests. As a initial model, we prepared speaker dependent model(SD), gender dependent and speaker independent model(GD), and speaker independent model(SI). SD was trained using speech data from target male speaker MHT. For GD, speech data from five male speakers in which target speaker MHT did not include was used. For SI, speech data from five male and four female speakers in which target speaker MHT did not include was used. Figure 6.1 shows the size of the resultant model.

The difference of quality of speech synthesized by using each initial model was evaluated by a pair comparison test. Preference scores are shown in Fig. 6.5.2. From these figure, it is seen that there is no difference of speech quality between initial model SD, GD and SI.

6.5.3 Speaking rate

Fig. 6.5.3 shows generated spectra for a Japanese sentence which is not included in the training data, setting ρ to -0.1, 0, 0.1. Only the part corresponding to the first phrase "/t-o-k-a-i-d-e-w-a/", which means "in a city" in English, is shown in this figure. From the figure, it can be seen that some parts such as stationary parts of vowels have elastic durations, and other parts such as explosives have fixed

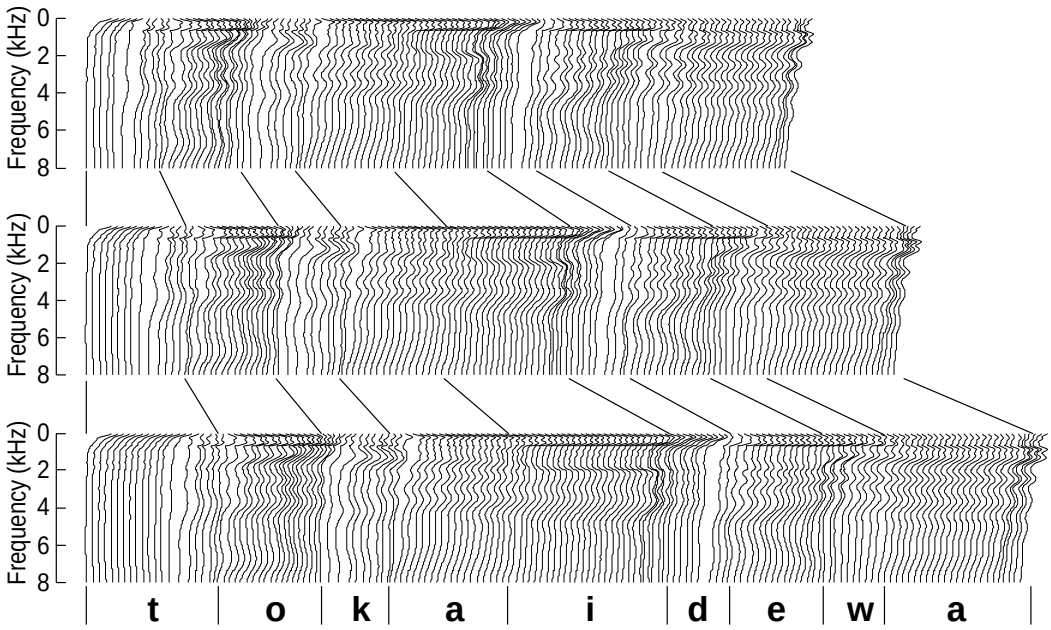


Figure 6.8: Generated spectra for an utterance "/t-o-k-a-i-d-e-w-a/" with different speaking rates (top : $\rho = -0.1$, middle : $\rho = 0$, bottom : $\rho = 0.1$).

durations. From informal listening tests, we found that synthetic speech had a good quality with natural timing. Furthermore, we confirmed that synthetic speech could keep natural timing even if its speaking rate was changed in some degree.

Chapter 7

Improvement of Synthesized Speech Quality

This chapter describes improvements on the excitation model of an HMM-based text-to-speech system. In our previous work, natural sounding speech can be synthesized from trained HMMs. However, it has a typical quality of "vocoded speech" since the system uses a traditional excitation model with either a periodic impulse train or white noise. In this thesis, in order to reduce the synthetic quality, a mixed excitation model used in MELP is incorporated into the system. Excitation parameters used in mixed excitation are modeled by HMMs, and generated from HMMs by a parameter generation algorithm in the synthesis phase. The result of a listening test shows that the mixed excitation model significantly improves quality of synthesized speech as compared with the traditional excitation model.

7.1 Introduction of Mixed Excitation Model

In the previous work [10], natural sounding speech can be synthesized from trained HMMs. However, synthesized speech has a typical quality of "vocoded speech" since the HMM-based TTS system used a traditional excitation model with either a periodic impulse train or white noise shown in Fig. 7.1. To overcome this problem, the excitation model should be replaced with more precise one.

For low bit rate narrowband speech coding at 2.4kbps, the mixed excitation linear predictive (MELP) vocoder has been proposed [37]. In order to reduce the synthetic quality and mimic the characteristics of natural human speech, this vocoder has the following capabilities:

- mixed pulse and noise excitation

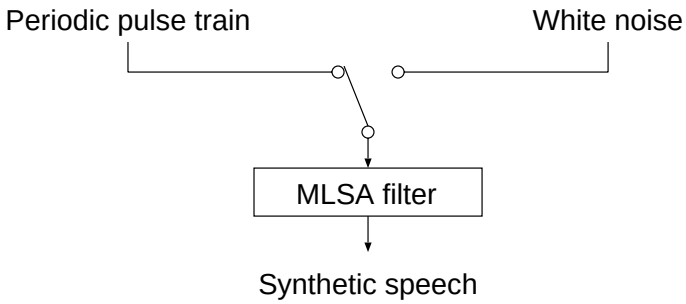


Figure 7.1: Traditional excitation model.

- periodic or aperiodic pulses
- pulse dispersion filter

The mixed excitation is implemented using a multi-band mixing model(Fig. 7.2), and can reduce the buzz of synthesized speech. Furthermore, aperiodic pulses and pulse dispersion filter reduce some of the harsh or tonal sound quality of synthesized speech. In recent years, the mixed excitation model of MELP has been applied not only to narrowband speech coding but also to wideband speech coder [38] and speech synthesis system [39].

In this thesis, mixed excitation model which is similar to the excitation model used in MELP is incorporated into the TTS system. Excitation parameters, i.e., F0, bandpass voicing strengths and Fourier magnitudes, are modeled by HMMs, and generated from trained HMMs in synthesis phase.

7.1.1 Mixed excitation

Analysis phase

In order to realize the mixed excitation model in the system, the following excitation parameters are extracted from speech data.

- F0
- bandpass voicing strengths
- Fourier magnitudes

In bandpass voicing analysis, the speech signal is filtered into five frequency bands, with passbands of 0–1000, 1000–2000, 2000–4000, 4000–6000, 6000–8000Hz [38]. Note that the TTS system deals with 16kHz sampling speech. The voicing strength in each band is estimated using normalized correlation coefficients around the pitch

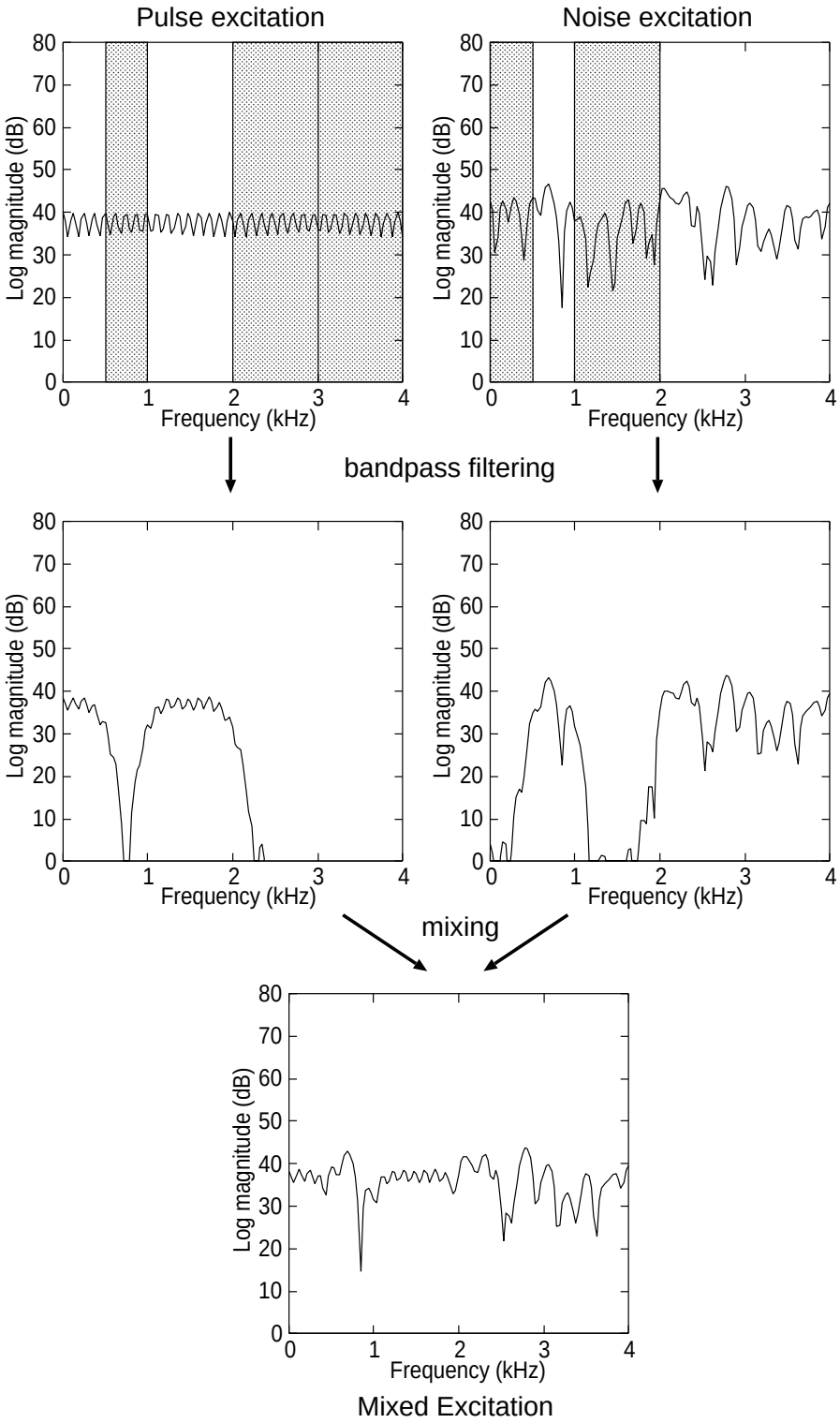


Figure 7.2: Multi-band mixing model.

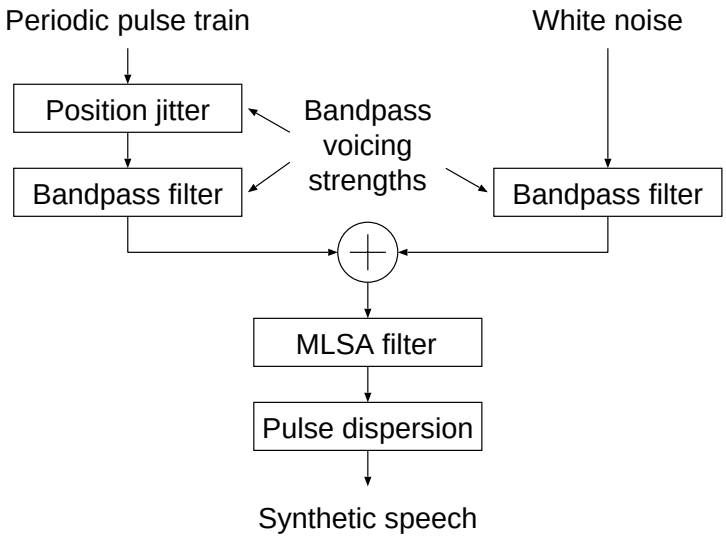


Figure 7.3: Mixed excitation model.

lag. The correlation coefficient at delay t is defined by

$$c_t = \frac{\sum_{n=0}^{N-1} s_n s_{n+t}}{\sqrt{\sum_{n=0}^{N-1} s_n s_n \sum_{n=0}^{N-1} s_{n+t} s_{n+t}}}, \tag{7.1}$$

where s_n and N represent the speech signal at sample n and the size of pitch analysis window, respectively. The Fourier magnitudes of the first ten pitch harmonics are measured from a residual signal obtained by inverse filtering.

Synthesis phase

A block diagram of the mixed excitation generation and speech synthesis filtering is shown in Fig. 7.3.

The bandpass filters for pulse train and white noise are determined from generated bandpass voicing strength. The bandpass filter for pulse train is given by the sum of all the bandpass filter coefficients for the voiced frequency bands, while the bandpass filter for white noise is given by the sum of the bandpass filter coefficients for the unvoiced bands. The excitation is generated as the sum of the filtered pulse and noise excitations. The pulse excitation is calculated from Fourier magnitudes using an inverse DFT of one pitch period in length. The pitch used here is adjusted by varying 25% of its position according to the periodic/aperiodic flag decided from the bandpass voicing strength. By the aperiodic pulses, the system mimics the erratic

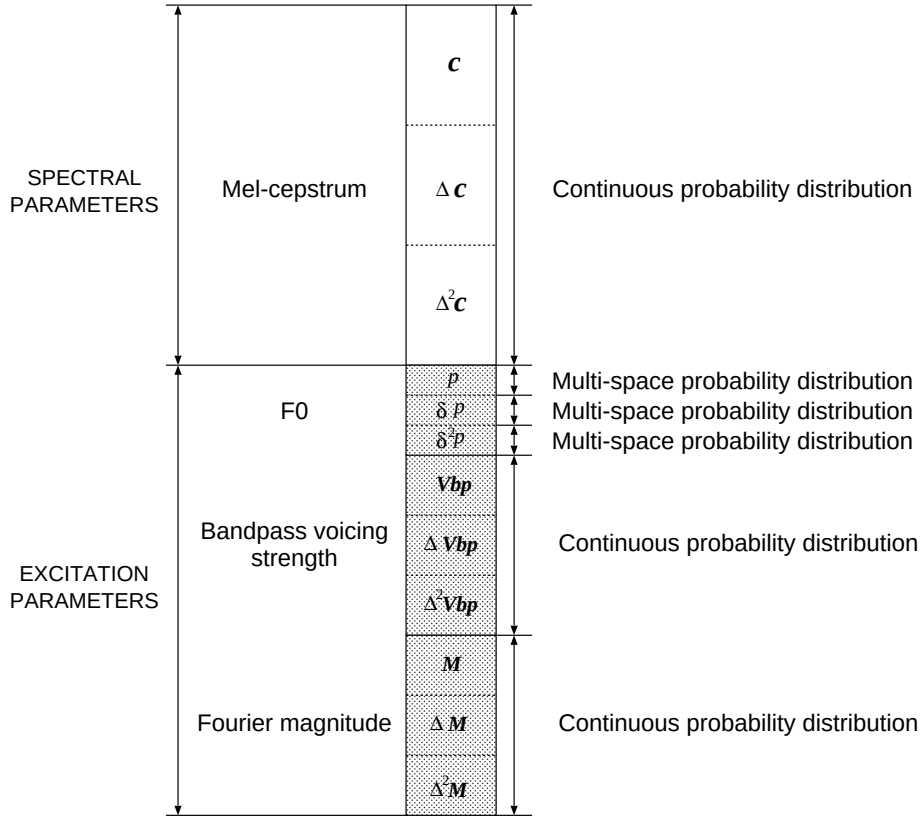


Figure 7.4: Structure of a feature vector modeled by HMM.

glottal pulses and reduces the tonal noise. The noise excitation is generated by a uniform random number generator. The obtained pulse and noise excitations are filtered and added together.

By exciting the MLSA filter [19], synthesized speech is generated from the mel-cepstral coefficients, directly. Finally, the obtained speech is filtered by a pulse dispersion filter which is a 130-th order FIR filter derived from a spectrally-flattened triangle pulse based on a typical male pitch period. The pulse dispersion filter can reduce some of the harsh quality of the synthesized speech.

7.1.2 Excitation parameter modeling

Feature vector

The structure of the feature vector is shown in Fig. 7.4. The feature vector consists of spectral and excitation parameters.

Mel-cepstral coefficients including zero-th coefficient and their delta and delta-delta

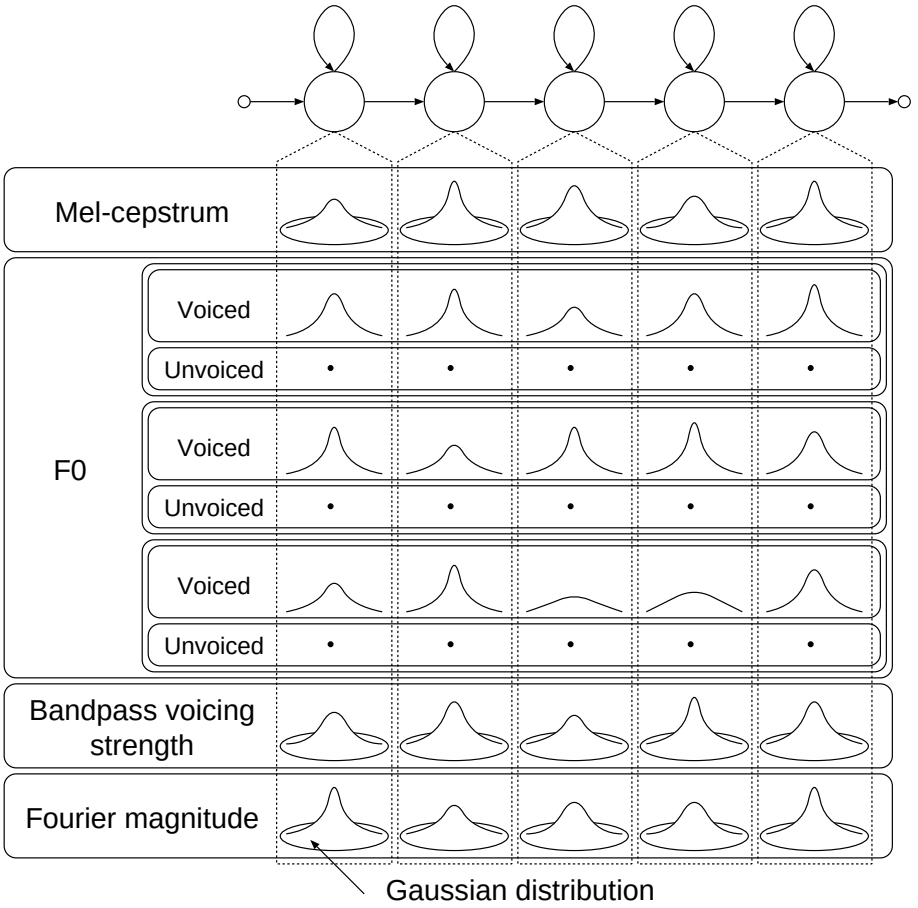


Figure 7.5: Structure of a HMM.

coefficients are used as spectral parameters. By using a mel-cepstral analysis technique [16] of an order of 24, mel-cepstral coefficients are obtained from speech signal windowed by a 25-ms Blackman window with a 5-ms shift.

Excitation parameters include F0 represented by log fundamental frequency ($\log f_0$), five bandpass voicing strengths, Fourier magnitudes of the first ten pitch harmonics, and their delta and delta-delta parameters.

Context dependent model

Feature vectors are modeled by 5-state left-to-right HMMs. Each state of an HMM has four streams for mel-cepstrum, F0, bandpass voicing strengths and Fourier magnitudes, respectively(Fig. 7.5). In each state, mel-cepstrum, bandpass voicing strengths and Fourier magnitudes are modeled by single diagonal Gaussian distributions, and F0 is modeled by the multi-space probability distribution [40].

Feature vectors are modeled by context dependent HMM taking account of contextual factors which affect spectral parameter and excitation parameter such as phone identity factors, stress-related factors and locational factors. Details of the contextual factors are shown in [10]. The trained context dependent HMMs are clustered using a tree-based context clustering technique based on MDL principle [36]. Since each of mel-cepstrum, F0, bandpass voicing strength, Fourier magnitude and duration has its own influential contextual factors, the distributions for each speech parameter are clustered independently, where state occupation statistics used for clustering are calculated from only the streams of mel-cepstrum and F0.

7.1.3 Excitation parameter generation

A context dependent label sequence is obtained by text analysis of input text, and a sentence HMM is constructed by concatenating context dependent phoneme HMMs according to the obtained label sequence. By using a speech parameter generation algorithm [41], mel-cepstrum, F0, bandpass voicing strength and Fourier magnitude are generated from the sentence HMM taking account of their respective dynamic feature statistics. Speech is synthesized from the obtained spectral and excitation parameters.

7.2 Incorporation of postfilter

Many speech coders, which include the MELP, attempted to improve synthesized speech quality by incorporating postfilter, and succeeded it. The our TTS system also incorporates the postfilter.

In Chapter 2, the synthesis filter $D(z)$ was realized using the MLSA filter. In order to realize the postfilter, first, a transfer function $\bar{D}(z)$ is defined. The transfer function $\bar{D}(z)$ is the same as $D(z)$ except that $c(1)$ is forced to be zero to compensate for the global spectral tilt. By setting $c(1) = 0$, the transfer function $\bar{D}(z)$ is written by

$$\bar{D}(z) = \exp \sum_{m=1}^{M} \bar{b}(m)\Phi_m(z) \tag{7.2}$$

$$\bar{b}(m) = \begin{cases} b(m) & 2 \leq m \leq M \\ -\alpha b(2) & m = 1 \end{cases} \tag{7.3}$$

We can realize the postfilter $\bar{D}^{\beta}(z)$ in the same manner as $\bar{D}(z)$, by multiplying $c(m)$ by β. The tunable parameter β control the amount of postfiltering.

The effect of the postfiltering is shown in Fig. 7.6 In Fig. 7.6, β is set to 0.5. From the figure, it can be observed that formant is emphasized by the postfiltering.

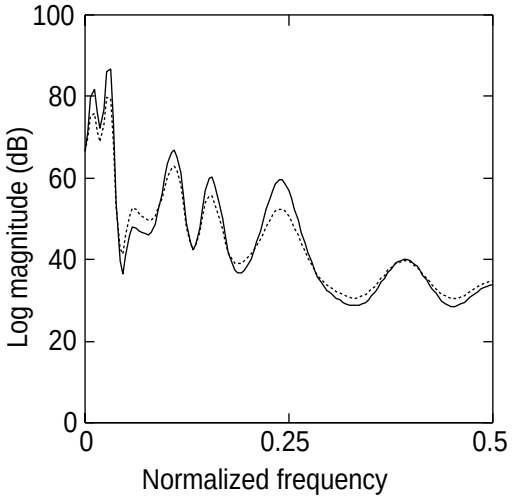


Figure 7.6: Effect of postfiltering(dots: before postfiltering, solid: after postfiltering $\beta = 0.5$).

7.3 Experiments

7.3.1 Excitation generation

Excitation parameters were generated from an HMM set trained using phonetically balanced 450 sentences of ATR Japanese speech database. The resulting decision trees for mel-cepstrum, F0, bandpass voicing strength, Fourier magnitude and state duration models had 934, 1055, 1651, 3745 and 1016 leaves in total, respectively. A part of each decision tree is shown in Appendix 9.2.

Examples of traditional excitation and mixed excitation are shown in Fig. 7.7, where the pulse dispersion filter was applied to mixed excitation. From the figure, it can be observed that the voiced fricative consonant "z" has both the periodic and aperiodic characteristics in the mixed excitation.

7.3.2 Effect of mixed excitation

The TTS system with mixed excitation model was evaluated. We compared traditional excitation and mixed excitation by a pair comparison test. In addition, effects of the Fourier magnitudes, aperiodic pulses and the pulse dispersion filter were evaluated.

The following five excitation models were compared:

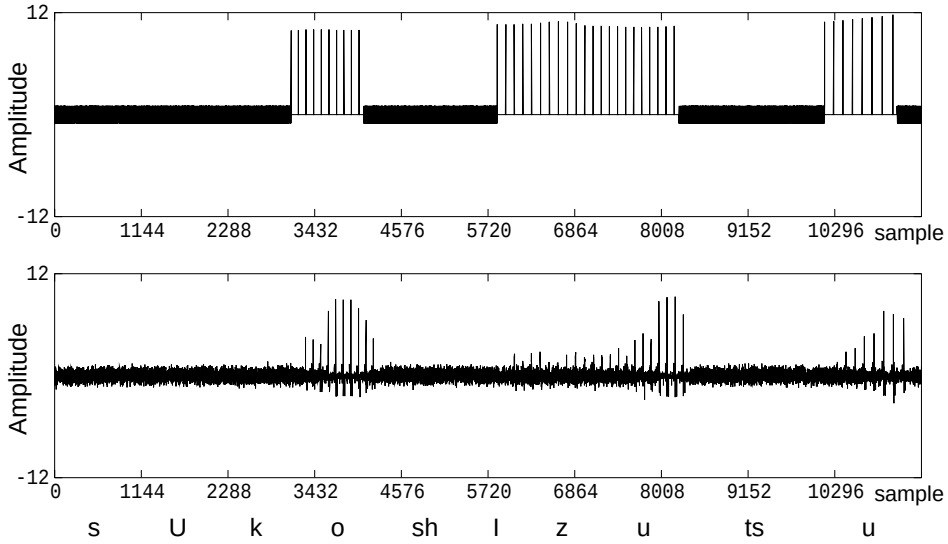


Figure 7.7: Example of generated excitation for phrase "sukoshizutsu." (top: traditional excitation, bottom: mixed excitation)

TE	:	traditional excitation
ME	:	mixed excitation
FM	:	**ME** + Fourier magnitudes
AP	:	**FM** + aperiodic pulses
PD	:	**AP** + pulse dispersion filter

The model **TE** is the traditional excitation model which generates either periodic pulse train or white noise. Each of models **ME**, **FM**, **AP** and **PD** is the mixed excitation model. In the model **ME**, pulse train was not calculated from Fourier magnitude, and the aperiodic pulse and the pulse dispersion filter were not applied. In the model **FM**, pulse excitation was calculated from Fourier magnitude. The model **AP** used aperiodic pulses, and the model **PD** used the pulse dispersion filter additionally. Eight subjects tested the five kinds of synthesized speech. Eight sentences were selected at random for each subjects from 53 sentences which were not included in the training data. Figure 7.8 shows preference scores. It can be seen that the mixed excitation model significantly improved the quality of synthetic speech. Although no additional gain was obtained by using Fourier magnitudes and aperiodic pulses, the additional use of pulse dispersion filter achieved further improvement in speech quality.

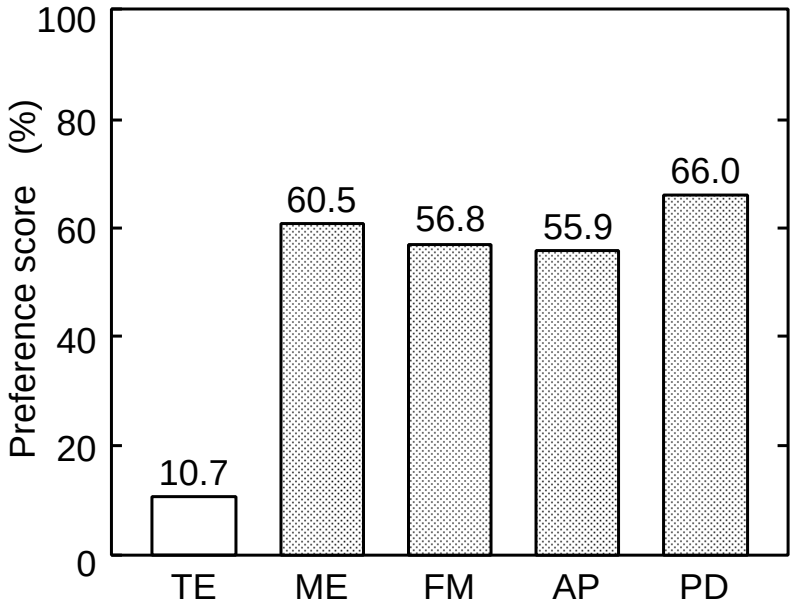


Figure 7.8: Comparison of traditional and mixed excitation models.

7.3.3 Effect of postfiltering

The effect of the postfiltering was evaluated by a pair comparison test. The following four samples were compared:

- conventional excitation model without postfiltering
  (NORMAL)
- conventional excitation model with postfiltering
  (NORMAL+POST)
- mixed excitation model without postfiltering
  (MIXED)
- mixed excitation model with postfiltering
  (MIXED+POST)

The mixed excitation model indicates **PD** in Section 7.3.

Figure 7.9 shows preference scores. It can be seen that the mixed excitation model with postfiltering significantly improved the quality of synthetic speech. Even the conventional excitation model with postfiltering also can improve the quality.

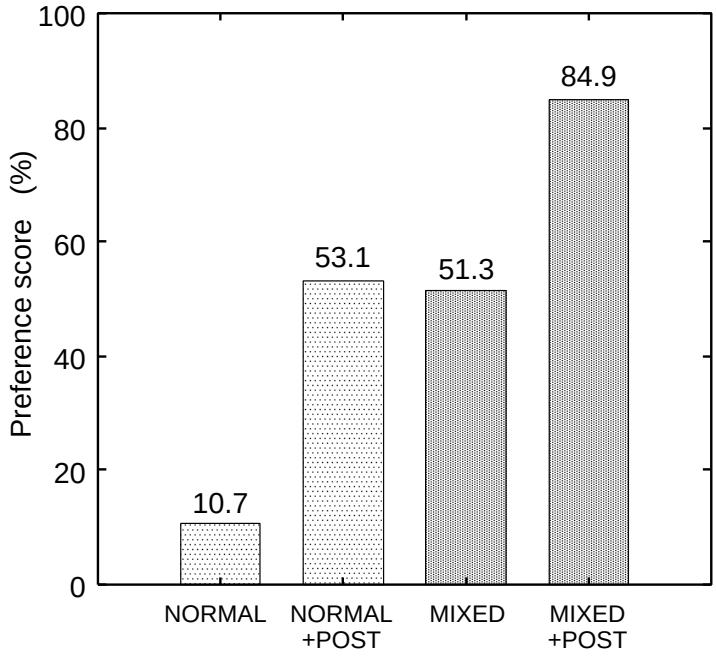


Figure 7.9: Effect of mixed excitation and postfiltering.

Chapter 8

Voice Conversion Technique: Speaker Interpolation

Although TTS systems which synthesize speech by concatenating speech units can synthesize speech with acceptable quality, they still cannot synthesize speech with various voice quality such as speaker individualities and emotions. To obtain various voice quality in text-to-speech synthesis systems based on the selection and concatenation of acoustical units, large amounts of speech data is required. However, it is difficult to collect, segment, and store these data. In this chapter, to synthesize speech with various voice characteristics, a TTS system based on speaker interpolation is proposed.

8.1 Overview

The proposed system[42] synthesizes speech with untrained speaker's voice quality by interpolating HMM parameters among some representative speakers' HMM sets. The idea of using speaker interpolation has been applied to voice conversion[43]. The proposed method differs from it in that each speech unit is modeled by an HMM, accordingly mathematically-well-defined statistical distance measures can be used for interpolating HMMs. As a result, the system can synthesize speech with various voice quality without large speech database in synthesis phase.

A block diagram of the TTS system based on speaker interpolation is shown in Fig. 8.1, which is almost equivalent to the previously proposed system except that multiple speaker's HMM sets are trained and a new speaker's HMM set is generated by interpolation among them. The procedure can be summarized as follows:

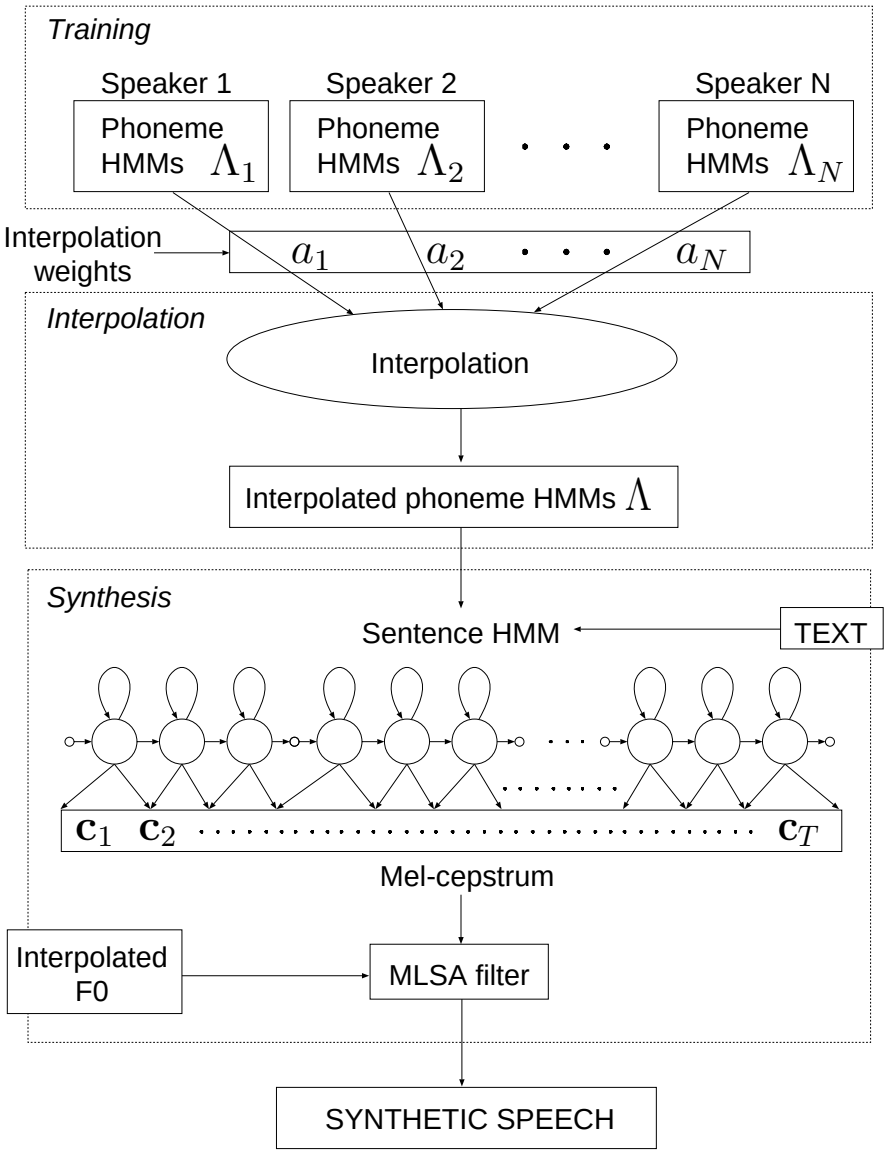


Figure 8.1: Block diagram of speech synthesis system with speaker interpolaiton.

1. Training representative HMM sets
   (a) Select several representative speakers $S_1,\ S_2,\ \ldots,\ S_N$ from speech database.
   (b) Obtain mel-cepstral coefficients from speech of the representative speakers by mel-cepstral analysis.
   (c) Train phoneme HMM sets $\Lambda_1,\ \Lambda_2,\ \ldots,\ \Lambda_N$ for $S_1,\ S_2,\ \ldots,\ S_N$, respectively, using mel-cepstral coefficients, and their deltas and delta-deltas.
2. Interpolation among representative HMM sets
   (a) Generate a new phoneme HMM set Λ by interpolating among the representative speakers' phoneme HMM sets $\Lambda_1,\ \Lambda_2,\ \ldots,\ \Lambda_N$ with an arbitrary interpolation ratio $a_1,\ a_2,\ \ldots,\ a_N$ based on a method described in the next section.
3. Speech synthesis from interpolated HMM
   (a) Convert the text to be synthesized into a phoneme sequence, and concatenate the interpolated phoneme HMMs according to the phoneme sequence.
   (b) Generate mel-cepstral coefficients from the sentence HMM by using speech parameter generation algorithm.
   (c) Synthesize speech from the generated mel-cepstral coefficients by using the MLSA (Mel Log Spectral Approximation) filter.

8.2 Speaker Interpolation

Figure 8.2 shows a space which represents speaker individuality. Representative speakers $S_1,\ S_2,\ \ldots,\ S_N$ are modeled by HMMs, $\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_N$, respectively. We assume that representative speaker's HMMs have the same topology (distributions could be tied). Under this assumption, interpolation among HMMs is equivalent to interpolation among output probability densities of corresponding states when state-transition probabilities are ignored. If we assume that each HMM state has a single Gaussian output probability density, the problem is reduced to interpolation among N Gaussian pdfs, $p_k(\boldsymbol{o}) = \mathcal{N}(\boldsymbol{o};\ \boldsymbol{\mu}_k, \boldsymbol{U}_k), \quad k = 1,\ 2,\ \ldots,\ N$, where $\boldsymbol{\mu}_k$ and $\boldsymbol{U}_k$ denote mean vector and covariance matrix, respectively, and $\boldsymbol{o}$ is the speech parameter vector.

We consider three methods to interpolate among pdfs as follows:

(a) When we define the interpolated pdf $p(\boldsymbol{o}) = \mathcal{N}(\boldsymbol{o};\ \boldsymbol{\mu}, \boldsymbol{U})$ as pdf of random variable

$$\boldsymbol{o} = \sum_{k=1}^{N} a_k \boldsymbol{o}_k, \tag{8.1}$$

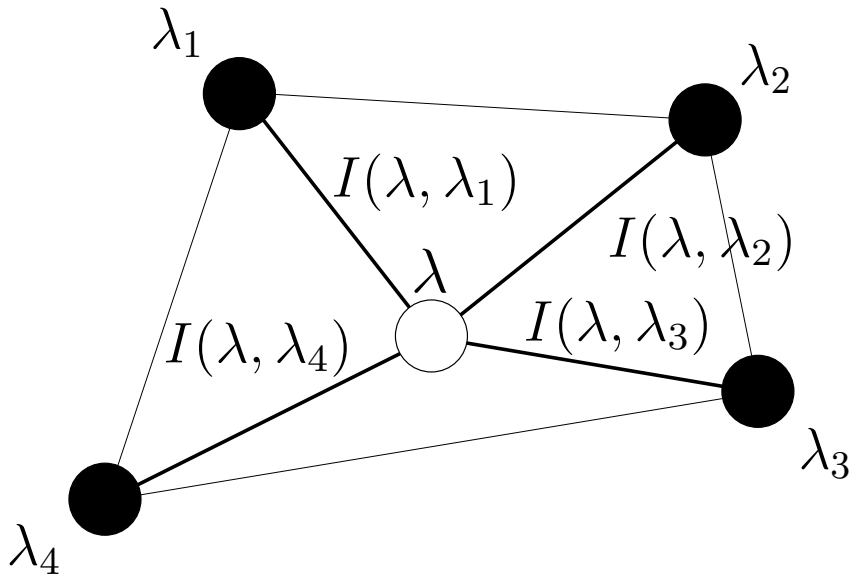


Figure 8.2: A space of speaker individuality modeled by HMMs.

where $\sum_{k=1}^{N} a_k = 1$, the mean $\boldsymbol{\mu}$ and variance $\boldsymbol{U}$ is calculated as follows:

$$\boldsymbol{\mu} = \sum_{k=1}^{N} a_k \boldsymbol{\mu}_k, \tag{8.2}$$

$$\boldsymbol{U} = \sum_{k=1}^{N} a_k^2 \boldsymbol{U}_k. \tag{8.3}$$

(b) We assume that mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{U}_k$ are trained by using γ_k feature vectors of speaker k. If the interpolated pdf p is trained by using feature vectors of N representative speakers, this pdf p is determined as

$$\boldsymbol{\mu} = \frac{\sum_{k=1}^{N} \gamma_k \boldsymbol{\mu}_k}{\gamma} = \sum_{k=1}^{N} a_k \boldsymbol{\mu}_k, \tag{8.4}$$

$$\begin{aligned} \boldsymbol{U} &= \frac{\sum_{k=1}^{N} \gamma_k \boldsymbol{U}_k}{\gamma} - \boldsymbol{\mu}\boldsymbol{\mu}' \\ &= \sum_{k=1}^{N} a_k (\boldsymbol{U}_k + \boldsymbol{\mu}_k \boldsymbol{\mu}_k') - \boldsymbol{\mu}\boldsymbol{\mu}' \end{aligned} \tag{8.5}$$

respectively, where $\gamma = \sum_{k=1}^{N} \gamma_k$ and $a_k = \gamma_k / \gamma$.

(c) We assume that the similarity between the interpolated speaker S and each representative speaker S_k can be measured by Kullback information measure between p and p_k. Then, for given pdfs $p_1, p_2, \ldots, p_N$ and weights $a_1, a_2, \ldots, a_N$, consider a problem to obtain pdf p which minimizes a cost function

$$\varepsilon = \sum_{k=1}^{N} a_k \, I(p, \, p_k), \tag{8.6}$$

that is, we can determine the interpolated pdf $p(\boldsymbol{o}) = \mathcal{N}(\boldsymbol{o}; \, \boldsymbol{\mu}, \boldsymbol{U})$ by minimizing ε with respect to $\boldsymbol{\mu}$ and $\boldsymbol{U}$, where the Kullback information measure can be written as

$$\begin{aligned} & I(p, \, p_k) \\ & \quad = \int_{-\infty}^{\infty} \mathcal{N}(\boldsymbol{o}; \, \boldsymbol{\mu}, \boldsymbol{U}) \, \log \frac{\mathcal{N}(\boldsymbol{o}; \, \boldsymbol{\mu}, \boldsymbol{U})}{\mathcal{N}(\boldsymbol{o}; \, \boldsymbol{\mu}_k, \boldsymbol{U}_k)} \, d\boldsymbol{o} \\ & \quad = \frac{1}{2} \Big\{ \log \frac{|\boldsymbol{U}_k|}{|\boldsymbol{U}|} + \\ & \quad \mathrm{tr} \left[\boldsymbol{U}_k^{-1} \{ (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})' + \boldsymbol{U} \} \right] + \mathbf{I} \Big\}. \end{aligned} \tag{8.7}$$

As a result, $\boldsymbol{\mu}$ and $\boldsymbol{U}$ are determined by

$$\boldsymbol{\mu} = \left(\sum_{k=1}^{N} a_k \boldsymbol{U}_k^{-1} \right)^{-1} \left(\sum_{k=1}^{N} a_k \boldsymbol{U}_k^{-1} \boldsymbol{\mu}_k \right), \tag{8.8}$$

$$\boldsymbol{U} = \left(\sum_{k=1}^{N} a_k \boldsymbol{U}_k^{-1} \right)^{-1}, \tag{8.9}$$

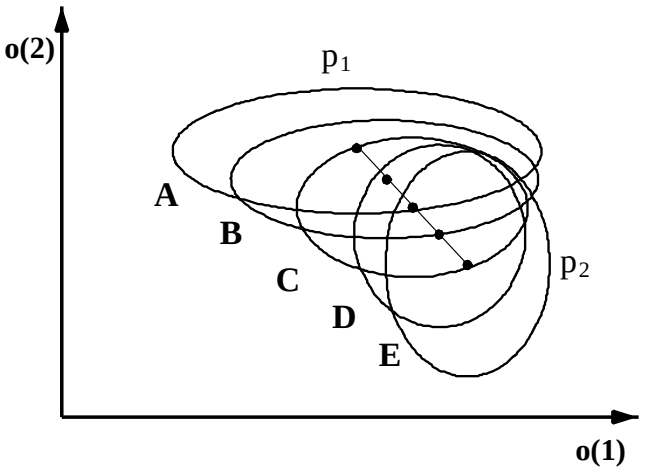
respectively. The derivation of (8), (9) is shown in Appendix A.

8.3 Simulation

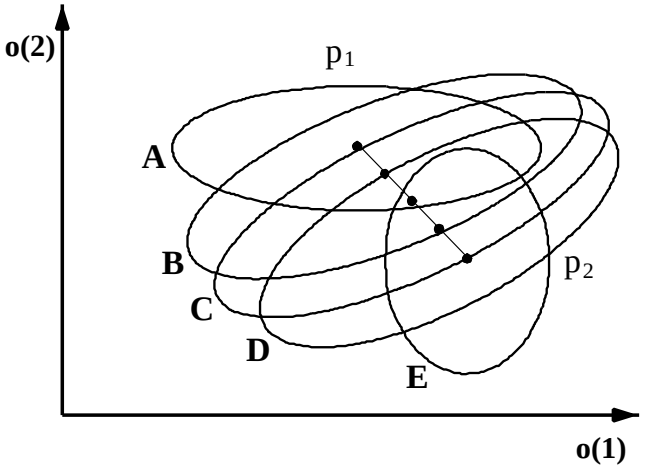
Table 8.1: Setting for simulating Japanese vowel "a"

	Formant	Center frequency (standard deviation)	Bandwidth
p_1 (male)	f1	750 Hz (100 Hz)	50 Hz
	f2	1100 Hz (130 Hz)	60 Hz
p_2 (female)	f1	1050 Hz (100 Hz)	50 Hz
	f2	1500 Hz (130 Hz)	60 Hz

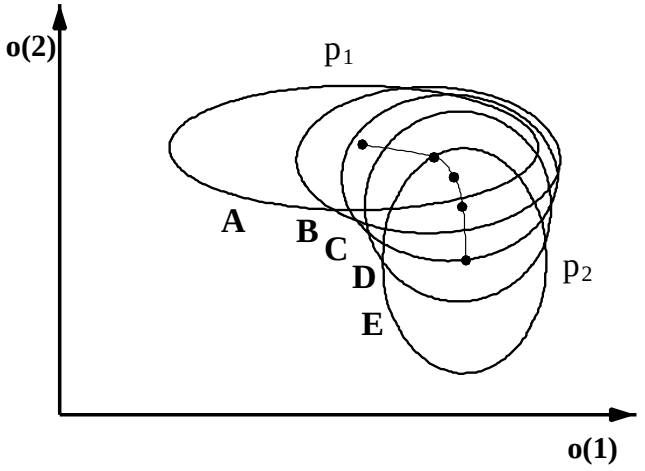
In order to check whether these methods generate interpolated distribution appropriately, we evaluated three methods based on simple simulations to avoid a time-consuming subjective evaluation.



(a)



(b)



(c)

Figure 8.3: Comparison between method (a), (b) and (c) with regard to interpolation between two multi-dimensional Gaussian distributions.

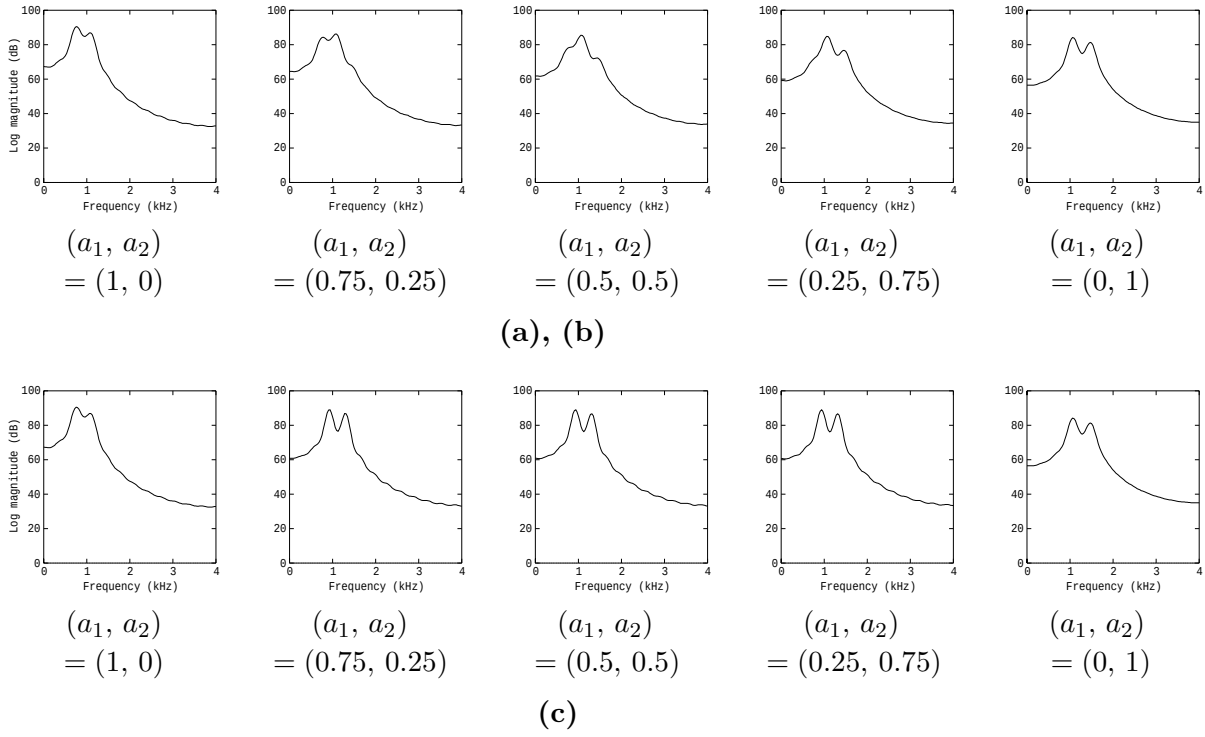


Figure 8.4: Comparison between method (a), (b) and (c) with regard to interpolation between two Gaussian distributions p_1 and p_2 with interpolation ratios A: $(a_1, a_2) = (1, 0)$, B: $(a_1, a_2) = (0.75, 0.25)$, C: $(a_1, a_2) = (0.5, 0.5)$, D: $(a_1, a_2) = (0.25, 0.75)$, E: $(a_1, a_2) = (0, 1)$.

Fig. 8.3 shows Gaussian distributions generated by interpolating between Gaussian distributions p_1 and p_2 with two-dimensional diagonal covariances. In this figure, each ellipse represents the contour of a Gaussian distribution, and each dot represents the mean vector of the distribution. From the figure, it can be seen that in methods **(a)** and **(b)** the interpolated mean vector is determined irrespective of covariances of representative distributions p_1 and p_2. On the other hand, the method **(c)** can interpolate between two distributions appropriately in the sense that the interpolated distribution p reflects the statistical information, i.e., covariances of p_1 and p_2.

By using three interpolation methods, we interpolated between multi-dimensional Gaussian distributions p_1 and p_2. Each distribution is calculated by using 1024 cepstral coefficient vectors which simulate Japanese vowel "a" uttered by male or female. Center frequencies, their standard deviations and bandwidths of the first and second formants were determined as shown in Tab. 8.1. Center frequencies are determined referring to Tab. 3.1 in [44] which shows ranges of male's and female's formant frequency. Standard deviations of center frequencies are estimated from

Tab. 3-5(a)(b) in [45], which shows changes in formant frequency of Japanese vowels according to difference of the preceding consonant, Bandwidths are estimated from Fig. 5.32(a) and Fig. 5.33(b) in [46], which show formant bandwidths and difference of male's and female's first formant bandwidth, respectively. Fig. 8.4 shows spectra which correspond to mean vectors of interpolated Gaussian distributions. It can be seen that the formant structure of spectra interpolated by the method **(a)** and **(b)** are collapsed. On the other hand, the spectra interpolated by the method **(c)** keep the formant structure. These results suggest that the method **(c)** is most appropriate in the three methods for interpolating among HMMs which model speech spectra, and we choose the method **(c)** for the subjective evaluation in the next section.

8.4 Experiments

By analyzing the result of the subjective evaluation of similarity using Hayashi's fourth method of quantification [47], we investigated whether the quality of synthesized speech from the interpolated HMM set is in between representative speakers'. In these experiments, we use two representative speakers, since when we use many speakers, combinations of stimuli increase exponentially in similarity test.

We used phonetically balanced 503 sentences from ATR Japanese speech database for training. Speech signals were sampled at 10 kHz and windowed by a 25.6 ms Hamming window with a 5 ms shift, and then mel-cepstral coefficients were obtained by mel-cepstral analysis. The feature vectors consisted of 16 mel-cepstral coefficients including the 0th coefficient, and their delta and delta-delta coefficients. Note that the 0th coefficient corresponds to logarithm of the signal gain. We used 5-state left-to-right triphone models with single Gaussian diagonal output distributions. Decision-tree based model clustering was applied to each set of triphone models, and the resultant set of tied triphone models had approximately 2,600 distributions.

We trained two HMM sets using speech data from a male speaker MHT and a female speaker FKN, respectively. By using the speech parameter generation algorithm, five different types of speech were synthesized from five HMM sets obtained by setting the interpolation ratio as (a_{MHT}, a_{FKN}) = (1, 0), (0.75, 0.25), (0.5, 0.5), (0.25, 0.75), (0, 1) (these sound files can be found in *http://kt−lab.ics.nitech.ac.jp/˜yossie/demo/speaker_inter.html*). The MLSA filter was excited by pulse train or white noise generated according to F0 contours. As the F0 contour for synthetic speech, we used a F0 contour obtained by linear interpolation between MHT's and FKN's F0 contours extracted from natural speech at a ratio of 1 : 1, where a one-to-one correspondence between MHT's and FKN's F0 contour was given by Viterbi alignment using spectral information. In each state, the time axis was linearly expanded or contracted. F0 contours were represented by logarithm of the frequency.

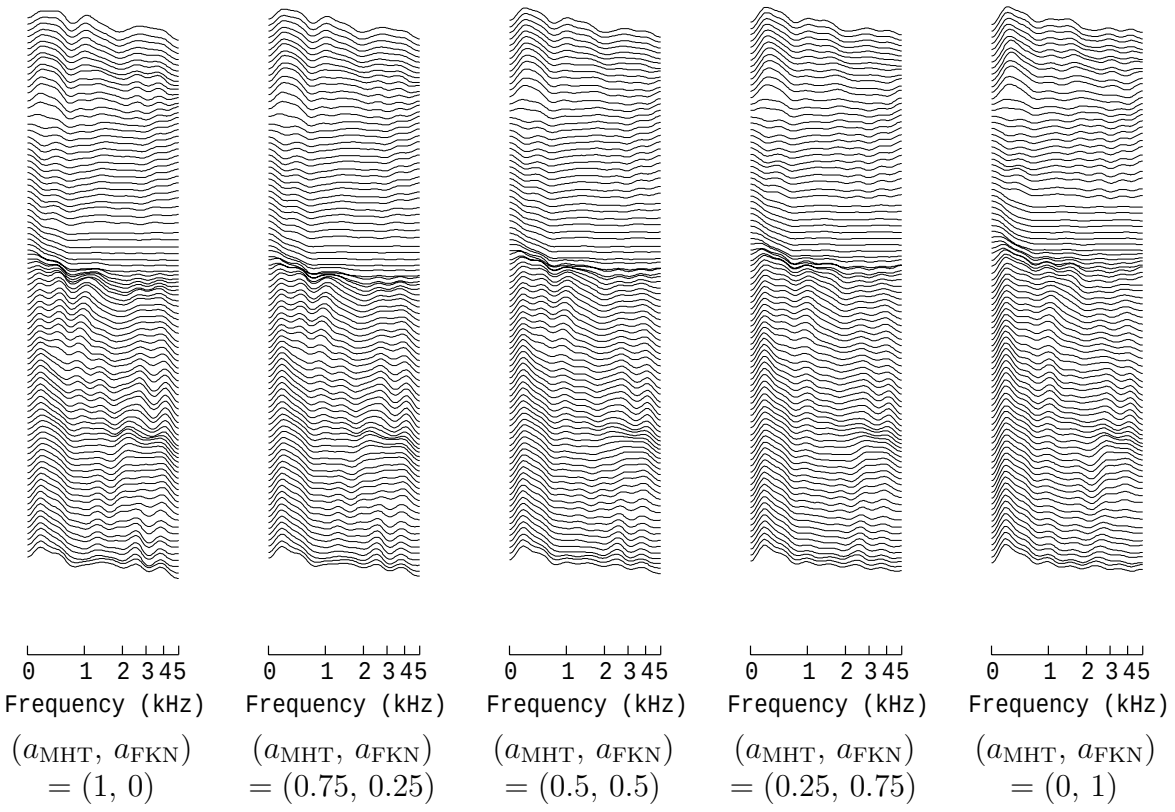


Figure 8.5: Generated spectra of the sentence "/n-i-m-o-ts-u/".

To observe only a change of spectrum, F0 contour was fixed for each sentence.

8.4.1 Generated Spectra

Fig. 8.5 shows spectra of a Japanese sentence "/n-i-m-o-ts-u/" generated from the triphone HMM sets. From the figure, it can be seen that spectra change smoothly from speaker MHT's to speaker FKN's according to the interpolation ratio.

8.4.2 Experiment of Similarity

In this experiment, two sentences (Appendix B), which were not included in the training data, were synthesized. Subjects were eight males. Stimuli consisted of two samples in five utterances which were synthesized with different interpolation ratios. Subjects were asked to rate the similarity of each pair into five categories ranging from "similar" to "dissimilar". From the results, we placed each sample in a space according to the similarities between the samples by using Hayashi's fourth method

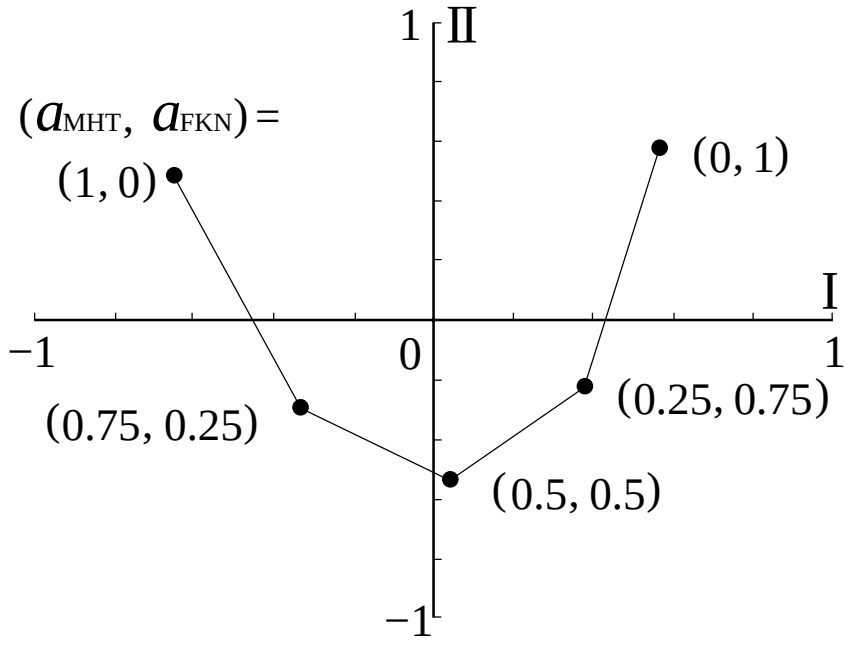


Figure 8.6: Subjective distance between samples.

of quantification.

Fig. 8.6 shows the relative similarity distance between stimuli. It is observed that the first axis roughly corresponds to gender. From this figure, it is seen that a pair of samples, whose interpolation ratios were close to each other, were placed closely on a two-dimensional space. This indicates that the voice quality of interpolated speaker changes smoothly between those of two representative speakers if interpolation ratio is linearly changed.

8.5 Discussion

In this thesis, we described an approach to voice quality conversion for an HMM-based text-to-speech synthesis system by interpolation among HMMs of representative speakers. As a result, the system can synthesize speech with various voice quality without large speech database in synthesis phase. From the results of experiments, we have seen that the quality of synthesized speech from the interpolated HMM set change smoothly from one male speaker's to the other female speaker's according to the interpolation ratio.

In this thesis, although we choose the method **(c)** based on the simulation, the

superiority of the method **(c)** should also be confirmed by the subjective evaluation. Although the proposed method can be applied to multiple representative speakers, the subjective evaluation was performed for two representative speakers. Thus, the subjective evaluation for the interpolation among multiple speakers is also our future work. We expect that the emotion (e.g., anger, sadness, joy) interpolation might be possible by replacing HMMs of representative speakers with those of representative emotions.

Chapter 9

Conclusions

In this chapter, we summarize the contributions of the thesis and suggest some direction for future work.

9.1 Original contribution revisited

The original contributions are summarized as follows. This summary is corresponded to the "original contribution" list given at the start of this thesis (Section 1.3)

- **Speech parameter generation using multi-mixture HMM.**
  In the previous works, single-mixture HMMs were used. However, the formant structure of spectrum corresponding to each mean vector might be vague since mean vector is the average of different speech spectra. In chapter 4, we proposed a parameter generation algorithm using multi-mixture HMMs. From the experimental result, we confirmed that the formant structure of the generated spectra get clearer with increasing mixtures.

- **Duration modeling for the HMM-based TTS system.**
  In chapter 5, we proposed duration modeling for HMM-based TTS system. Duration models are constructed taking account of contextual factors that affect durations. From informal listening tests, we found that synthetic speech had a good quality with natural timing. Furthermore, we confirmed that synthetic speech could keep natural timing even if its speaking rate was changed in some degree.

- **Simultaneous modeling of spectrum, F0 and duration.**
  In chapter 5, we described an HMM-based speech synthesis system in which

spectrum, F0 and state duration are modeled simultaneously in a unified framework of HMM. As a result, it is possible that statistical voice conversion techniques (e.g., speaker adaptation technique, speaker interpolation technique) are applied to the proposed TTS system.

- **Training of context dependent HMM using MDL principle.**
  In chapter 5, the distributions for spectral parameter, F0 parameter and the state duration are clustered independently by using a decision-tree based context clustering technique based on the MDL principle. By the MDL principle, decision tree is set appropriate size taking account of amount of data.

- **F0 parameter generation using dynamic features.**
  We confirmed the effect of dynamic feature in the F0 parameter generation in chapter 6. By taking account of dynamic features, very smooth and natural F0 pattern can be generated. From the listening test, synthesized speech, whose spectra and F0 pattern are generated with dynamic features, is improved than speech synthesized without dynamic features.

- **Automatic training of the HMM-based TTS system.**
  The HMM-based TTS system can be constructed automatically using appropriate initial model and speech data without label boundary information. In chapter 6, actually we did automatic system construction, and confirmed that synthesized speech has the same quality with the case of using speech data with label boundary information.

- **Improvement quality of synthesized speech by incorporating mixed excitation model and postfilter into the HMM-based TTS system.**
  There are many approaches to the improvement of the speech quality for the HMM-based TTS system. As one of approaches, we tried to incorporate the mixed excitation model and the postfilter to the system in chapter 7. As a result, the quality of the synthesized speech could be significantly improved.

- **Voice conversion using speaker interpolation technique.**
  For the purpose of synthesizing speech with various voice characteristics such as speaker individualities and emotions, we proposed the TTS system based on speaker interpolation in chapter 8. The proposed system synthesizes speech with untrained speaker's voice quality by interpolating HMM parameters among some representative speakers' HMM sets. Listening tests show that the proposed algorithm successfully interpolates between representative speakers in the case where two representative HMM sets are trained by a male and a female speakers' speech data, respectively; the quality of synthesized speech is in between the male and female speakers', and can be gradually changed from one's to the other's according to interpolation ratio.

9.2 Future works

There are some works as far the proposed TTS system. For example, The following works are considered:

- **Improvement of synthesized speech quality.**
  In Chapter 7, the mixed excitation and postfilter were incorporated to the system, and they could improve the speech quality. However, this approach is used in the field of speech coding in which it is purpose that high quality speech are obtained with small amount of bit rate. It is considered that there is some approaches with a large number of speech parameters to improvement of the speech quality. For example, STRAIGHT[48] is a high-quality analysis-synthesis method and offers high flexibility in parameter manipulation with no further degradation. If analysis-synthesis method such as STRAIGHT can be incorporated to the our TTS system, it is considered that the synthesized speech quality is improved.

- **Emotional speech synthesis** In Chapter 8, we could change speaker individualities of the synthesized speech using speaker interpolation technique. Further, we expect that the emotion (e.g., anger, sadness, joy) interpolation might be possible by replacing HMMs of representative speakers with those of representative emotions.

Bibliography

[1] A. Ljolje, J. Hirschberg and J. P. H. van Santen, "Automatic speech segmentation for concatenative inventory selection," Progress in Speech Synthesis, ed. J. P. H. van Santen, R. W. Sproat, J. P. Olive and J. Hirschberg, Springer-Verlag, New York, 1997.

[2] R. E. Donovan and P. C. Woodland, "Automatic speech synthesiser parameter estimation using HMMs," Proc. of ICASSP, pp.640–643, 1995.

[3] R. E. Donovan and P. C. Woodland, "Improvements in an HMM-Based Synthesizer," Proc. of EUROSPEECH, pp.573–576, 1995.

[4] H. Hon, A. Acero, X. Huang, J. Liu and M. Plumpe, "Automatic generation of synthesis units for trainable text-to-speech synthesis," Proc. of ICASSP, pp.293–306, 1998.

[5] X. Huang, A. Acero, H. Hon, Y. Ju, J. Liu, S. Meredith and M. Plumpe, "Recent improvements on Microsoft's trainable text-to-speech system -Whistler," Proc. of ICASSP, pp.959–962, 1997.

[6] R. E. Donovan and E. M. Eide, "The IBM Trainable Speech Synthesis System," Proc. of ICSLP, vol.5, pp.1703–1706, Nov. 1998.

[7] A. Falaschi, M. Giustiniani and M. Verola, "A hidden Markov model approach to speech synthesis," Proc. of EUROSPEECH, pp.187–190, 1989.

[8] M. Giustiniani and P. Pierucci, "Phonetic ergodic HMM for speech synthesis," Proc. of EUROSPEECH, pp.349–352, 1991.

[9] T. Masuko, K. Tokuda, T. Kobayashi and S. Imai, "Speech synthesis from HMMs using dynamic features," Proc. of ICASSP, pp.389–392, 1996.

[10] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Simultaneous Modeling of Spectrum, Pitch and Duration in HMM-Based Speech Synthesis," Proc. of EUROSPEECH, S11.PO2.16, vol.5, pp.2347–2350, Sep. 1999.

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

[12] 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. of EUROSPEECH, pp.757–760, 1995.

[13] T. Masuko, K. Tokuda, T. Kobayashi and S. Imai, "Voice characteristics conversion for HMM-based speech synthesis system," Proc. of ICASSP, vol.3, pp.1611-1614, 1997.

[14] M. Tamura, T. Masuko, K. Tokuda and T. Kobayashi, "Speaker Adaptation for HMM-based Speech Synthesis System Using MLLR," Proc. of The Third ESCA/COCOSDA workshop on Speech Synthesis, pp.273–276, Dec. 1998.

[15] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Speaker Interpolation in HMM-Based Speech Synthesis System," Proc. of EUROSPEECH, Th4C.5, vol.5, pp.2523–2526, 25 Sep. 1997.

[16] T. Fukada, K. Tokuda, T. Kobayashi and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," Proc. of ICASSP, vol.1, pp.137–140, 1992.

[17] A.V. Oppenheim and R.W. Schafer, "Discrete-time signal processing," Prentice-Hall, Englewood Cliffs, N.J., 1989.

[18] S. Imai and C. Furuichi, "Unbiased estimator of log spectrum and its application to speech signal processing," Proc. of EURASIP, pp.203-206, Sep. 1988.

[19] S. Imai, "Cepstral analysis synthesis on the mel frequency scale," Proc. of ICASSP, pp.93–96, Feb. 1983.

[20] T. Kobayashi and S. Imai, "Complex Chebyshev approximation for IIR digital filters using an iterative WLS technique," Proc. of ICASSP, pp.377–380, Apr. 1990.

[21] L.A.Liporace, "Maximum Likelihood Estimation for Multivariate Observations of Markov Sources," IEEE Trans. Information Theory, IT-28, 5, pp.729–734, 1982.

[22] B.-H. Juang, "Maximum-likelihood estimation for mixture multivariate stachastic observations of Markov chains," AT&T Technical Journal, vol.64, no.6, pp.1235–1249, 1985.

[23] U. Jensen, R. K. Moore, P. Dalsgaard, and B Lindberg, "Modeling intonation contours at the phrase level using continuous density hidden Markov models," Computer Speech and Language, vol.8, no.3, pp.247–260, Aug. 1994.

[24] G. J. Freij and F. Fallside, "Lexical stress recognition using hidden Markov models," Proc. of ICASSP, pp.135–138, 1988.

[25] K. Ross and M. Ostendorf, "A dynamical system model for generating F_0 for synthesis," Proc. of ESCA/IEEE Workshop on Speech Synthesis, pp.131–134, 1994.

[26] M. Nishimura and K. Toshioka, "HMM-based speech recognition using multi-dimensional multi-labeling," Proc. of ICASSP, pp.1163–1166, 1987.

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

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

[29] W. Tachiwa and S. Furui, "A study of speech synthesis using HMMs," Proc. of Spring Meeting of Acoustical Society of Japan, pp.239–240, Mar. 1999 (in Japanese).

[30] S. E. Levinson, "Continuously Variable Duration Hidden Markov Models for Speech Analysis," Proc. of ICASSP, pp.1241–1244, 1986.

[31] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Duration Modeling in HMM-based Speech Synthesis System," Proc. of ICSLP, vol.2, pp.29–32, Nov. 1998.

[32] J. J. Odell, "The Use of Context in Large Vocabulary Speech Recognition," PhD dissertation, Cambridge University, 1995.

[33] N. Miyazaki, K. Tokuda, T. Masuko and T. Kobayashi, "A Study on Pitch Pattern Generation using HMMs Based on Multi-space Probability Distributions," Technical Report of IEICE, SP98-12, Apr. 1998(in Japanese).

[34] H. J. Nock, M. J. F. Gales and S. J. Young, "A Comparative Study of Methods for Phonetic Decision-Tree State Clustering," Proc. of EUROSPEECH, pp.111–115, 1997.

[35] W. Chou and W. Reichl, "Decision Tree State Tying Based on Penalized Bayesain Information Criterion," Proc. of ICASSP, pp.345–348, 1999.

[36] K. Shinoda and T. Watanabe, "Speaker Adaptation with Autonomous Model Complexity Control by MDL Principle," Proc. of ICASSP, pp.717–720, May 1996.

[37] A. V. McCree and T. P. Barnwell III, "A mixed excitation LPC vocoder model for low bit rate speech coding," IEEE Trans. Speech and Audio Processing, vol.3, no.4, pp.242–250, Jul. 1995.

[38] W. Lin, S. N. Koh and X. Lin, "Mixed excitation linear prediction coding of wideband speech at 8kbps" Proc. of ICASSP, vol.2, pp.1137–1140, Jun. 2000.

[39] N. Aoki, K. Takaya, Y. Aoki and T. Yamamoto, "Development of a rule-based speech synthesis system for the Japanese language using a MELP vocoder," IEEE Int. Sympo. on Intelligent Signal Processing and Communication Systems, pp.702–705, Nov. 2000.

[40] K. Tokuda, T. Masuko, N. Miyazaki and T. Kobayashi, "Hidden Markov Models Based on Multi-Space Probability Distribution for Pitch Pattern Modeling," Proc. of ICASSP, pp.229–232, May 1999.

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

[42] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, "Speaker Interpolation for HMM-based Speech Synthesis System," The Journal of the Acoustical Society of Japan(E), vol.21, no.4, pp.199–206, Jul. 2000.

[43] N. Iwahashi and Y. Sagisaka, "Speech Spectrum Conversion Based on Speaker Interpolation and Multi-functional Representation with Weighting by Radial Basis Function Networks," Speech Communication, 16, pp.139–151, 1995.

[44] S. Saito and K. Nakata, "Fundamentals of Speech Signal Processing," Ohmsha, Ltd. pp.36–38, 1981 (in Japanese).

[45] O. Fujimura, "Speech Science(Onsei Kagaku)," University of Tokyo Press, pp.213–218, 1972 (in Japanese).

[46] J.L. Flanagan, "Speech Analysis, Synthesis, and Perception," 2nd Edition, Springer-Verlag, pp.181–184, 1972.

[47] C. Hayashi, "Recent theoretical and methodological developments in multidimensional scaling and its related method in Japan," Behaviormetrika, vol.18, pp.1095, 1985.

[48] H. Kawahara, "Speech representation and transformation using adaptive interpolation of weighted spectrum: Vocoder revisited," Proc. of ICASSP, vol.2, pp.1303–1306, 1997.

List of Publications

Journal Papers

Toshiaki Fukada, **Takayoshi Yoshimura** and Yoshinori Sagisaka, "Automatic generation of multiple pronunciations based on neural networks," Speech Communication, vol.27, no.1, pp.63–73, Sep. 1999

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Speaker interpolation for HMM-based speech synthesis system," The Journal of the Acoustical Society of Japan(E), vol.21, no.4, pp199–206, Jul. 2000

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Simultaneous modeling of spectrum, pitch and state duration in HMM-based speech synthesis," *IEICE Trans. of Inf. & Syst.*, vol.J83-D-II,no.11,pp.2099–2107,Nov. 2000(in Japanese)

International Conference Proceedings

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Speaker interpolation in HMM-based speech synthesis system," Proc. of EUROSPEECH, vol.5, pp.2523–2526, Sep. 1997

Toshiaki Fukada, **Takayoshi Yoshimura** and Yoshinori Sagisaka, "Automatic

generation of multiple pronunciations based on neural networks and language statistics," Proc. of ESCA workshop on modeling pronunciation variation for automatic speech recognition, pp.41–46, 1998

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Duration modeling in HMM-based speech synthesis system," Proc. of ICSLP, vol.2, pp.29–32, Nov. 1998

Toshiaki Fukada, **Takayoshi Yoshimura** and Yoshinori Sagisaka, "Neural network based pronunciation modeling with applications to speech recognition," Proc. of ICSLP, vol.3, pp.723–726, Nov. 1998

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," Proc. of EUROSPEECH, vol.5, pp.2347–2350, Sep. 1999

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," Proc. ICASSP, vol.3, pp.1315–1318, June 2000

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, "Mixed excitation for HMM-based speech synthesis," Proc. of EUROSPEECH, vol.3, pp.2263–2266, Sep. 2001

Technical Reports

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "State duration modeling for HMM-based speech synthesis," *Technical Report of IEICE*, DSP98-85, vol.98, no.262, pp.45–50, Sep. 1998(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Simultaneous modeling of spectrum, pitch and state duration in HMM-based speech synthesis," *Technical Report of IEICE*, SP99-59, vol.99, no.255, pp.33–38, Aug. 1999(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Introduction of mixed excitation model and postfilter to HMM-based speech synthesis," *Technical Report of IEICE*, SP2001-63, vol.101, no.325, pp.17-22, Sep. 2001(in Japanese)

Atsushi Sawabe, Kengo Shichiri, **Takayoshi Yoshimura**, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Application of eigenvoice technique to spectrum and pitch pattern modeling in HMM-based speech synthesis," *Technical Report of IEICE*, SP2001-72, vol.101, no.325, pp.65-72, Sep. 2001(in Japanese)

Domestic Conference Proceedings

Takayoshi Yoshimura, Tohru Takahashi, Keiichi Tokuda and Tadashi Kitamura, "Word recognition using two-dimensional mel-cepstrum," *Tokai Sec. Joint Conf. of Seven Institutes of Electrical and Related Engineers '97*, pp250, Oct. 1996(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Speaker interpolation in HMM-based speech synthesis sys-

tem," *ASJ Autumn Meeting '97*, 1-P-17, I, pp.337–338, Sep. 1997(in Japanese)

Toshiaki Fukada, **Takayoshi Yoshimura** and Yoshinori Sagisaka, "Automatic generation of multiple pronunciations based on neural networks and language statistics," *ASJ Spring Meeting '98*, 2-Q-30, I, pp.179–180, Mar. 1998(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "State duration modeling for HMM-based speech synthesis," *ASJ Autumn Meeting '98*, 1-2-8, I, pp.189–190, Sep. 1998(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Spectrum, pitch and state duration modeling for HMM-based speech synthesis," *ASJ Spring Meeting '99*, 2-3-8, I, pp.241–242, Mar. 1999(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Effects of dynamic feature in HMM-based pitch pattern generation," *ASJ Autumn Meeting '99*, 1-3-16, I, pp.215–216, Sep. 1999(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Evaluation of speech parameter generation from HMM based on maximum likelihood criterion," *ASJ Spring Meeting '2000*, 1-7-7, I, pp.209–210, Mar. 2000(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Automatic construction of HMM-based speech synthesis system," *ASJ Autumn Meeting '2000*, 1-Q-2, I, pp.233–234, Sep. 2000(in Japanese)

Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "A study on Excitation model for HMM-based speech

synthesis," *ASJ Spring Meeting '2001*, 2-6-8, I, pp.297–298, Mar. 2001(in Japanese)

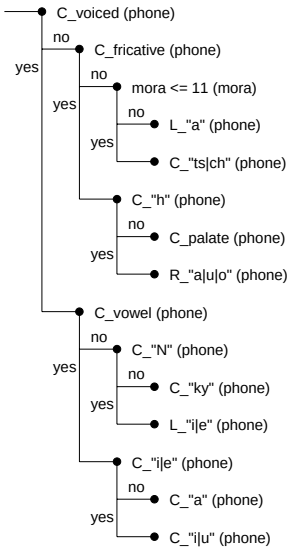
Atsushi Sawabe, Kengo Shichiri, **Takayoshi Yoshimura**, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Speech synthesis using triphone based on eigenvoices," *ASJ Spring Meeting '2001*, 2-6-9, I, pp.299–300, Mar. 2001(in Japanese)

Atsushi Sawabe, Kengo Shichiri, **Takayoshi Yoshimura**, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "Pitch modeling in eigenvoices-based speech synthesis," *ASJ Autumn Meeting '2001*, 3-2-6, I, pp.315–316, Oct. 2001(in Japanese)

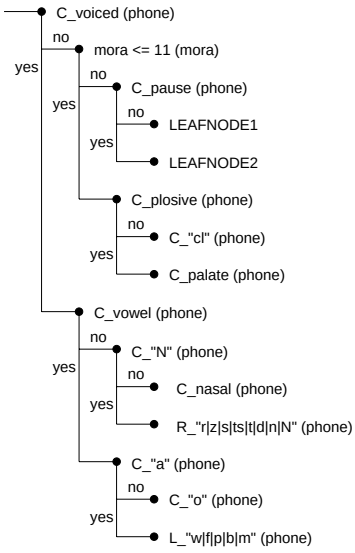
Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi and Tadashi Kitamura, "A study on improvement of speech quality for HMM-based speech synthesis," *ASJ Autumn Meeting '2001*, 1-P-8, I, pp.371-372, Oct. 2001(in Japanese)

Appendix A

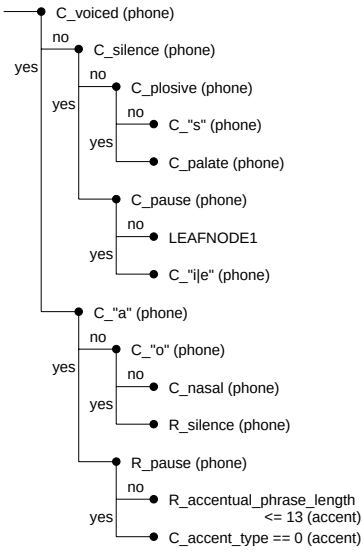
Examples of decision trees constructed by using the MDL principle.



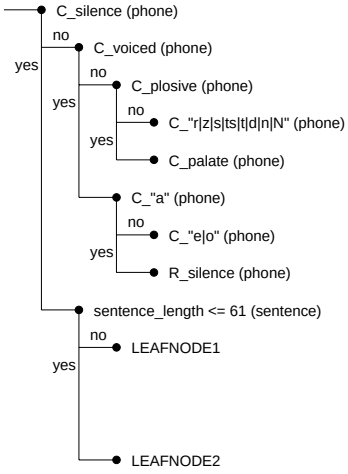
for state 1



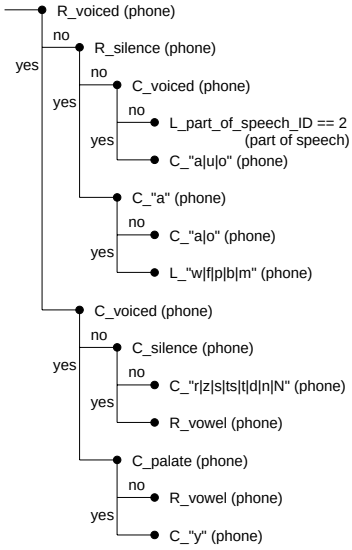
for state 2



for state 3



for state 4



for state 5

Figure A.1: Examples of decision trees for mel-cepstrum.

Figure A.2: Examples of decision trees for F0.

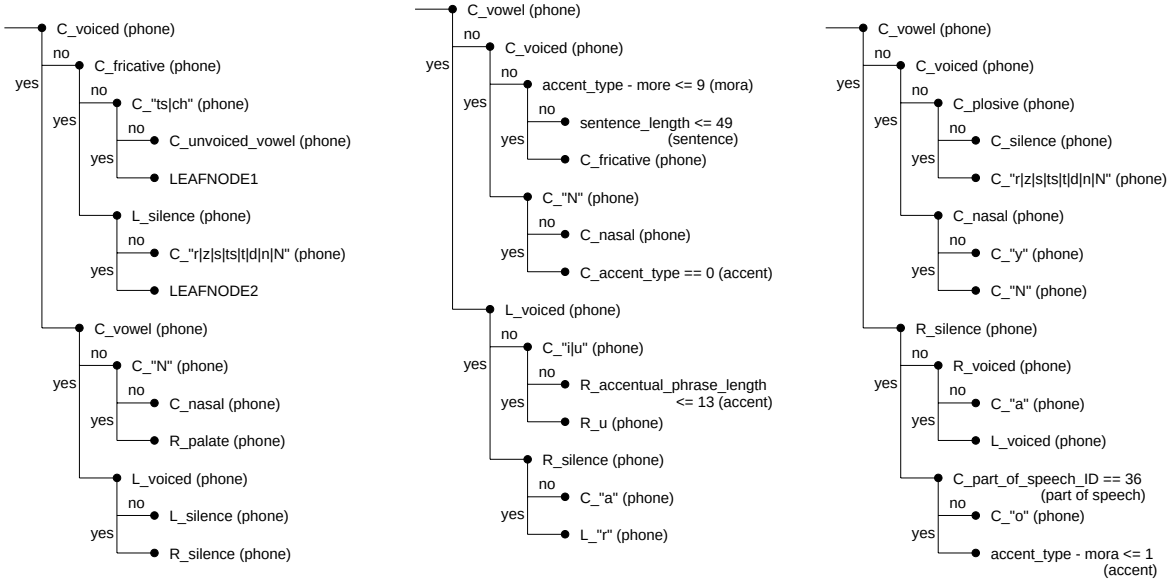
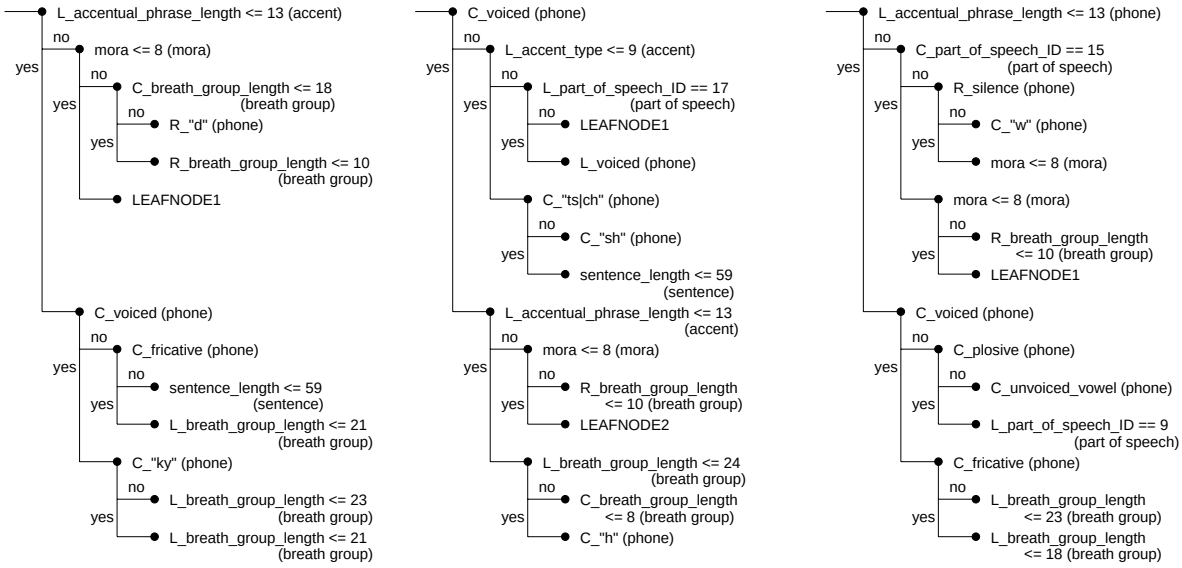
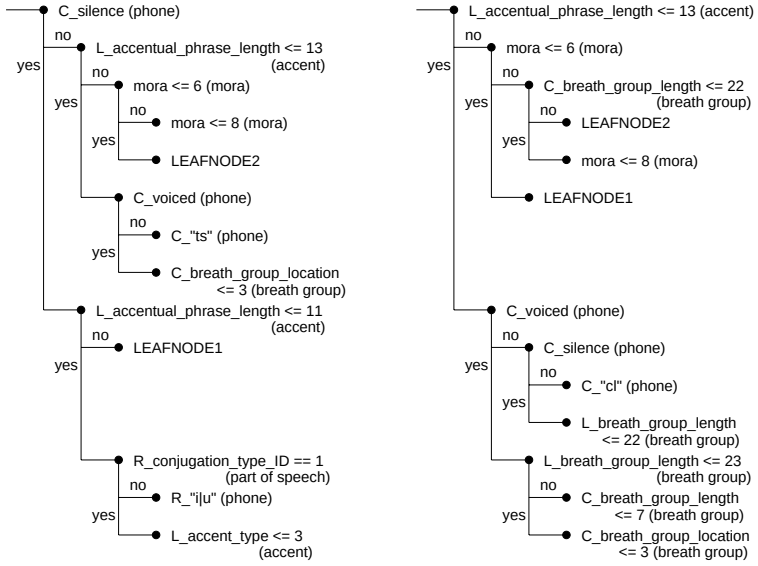


Figure A.3: Examples of decision trees for bandpass voicing strength.



for state 1 for state 2 for state 3



for state 4 for state 5

Figure A.4: Examples of decision trees for Fourier magnitude.

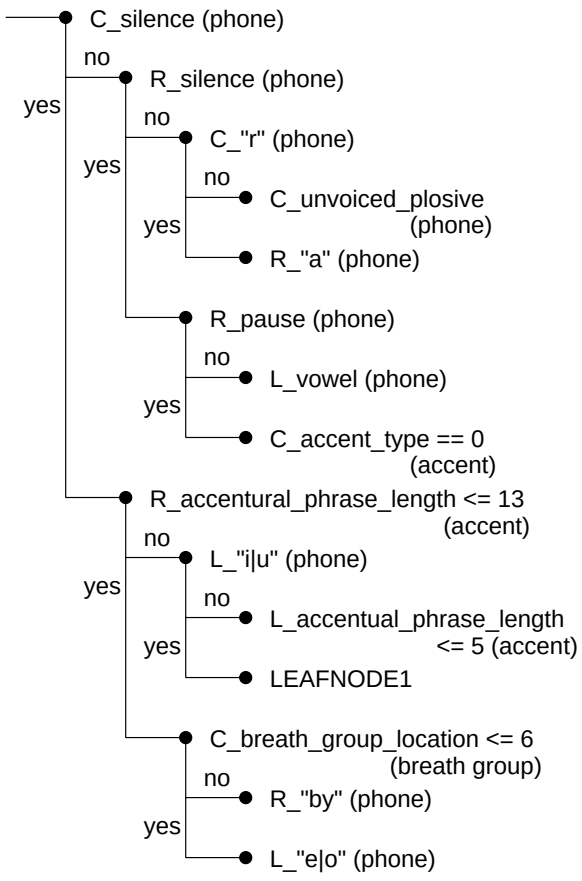


Figure A.5: Examples of decision trees for duration.