SPEECH PROSODY GENERATION FOR TEXT-TO-SPEECH SYNTHESIS BASED ON GENERATIVE MODEL OF F_0 CONTOURS

Kento Kadowaki[1], *Tatsuma Ishihara*[1], *Nobukatsu Hojo*[1], *Hirokazu Kameoka*[1,2]

[1]Graduate School of Information Science and Technology, The University of Tokyo, Japan
[2] NTT Communication Science Laboratories, NTT Corporation, Japan
{kadowaki,ishihara,hojo,kameoka}@hil.t.u-tokyo.ac.jp

Abstract

This paper deals with the problem of generating the fundamental frequency (F_0) contour of speech from a text input for text-to-speech synthesis. We have previously introduced a statistical model describing the generating process of speech F_0 contours, based on the discrete-time version of the Fujisaki model. One remarkable feature of this model is that it has allowed us to derive an efficient algorithm based on powerful statistical methods for estimating the Fujisaki-model parameters from raw F_0 contours. To associate a sequence of the Fujisaki-model parameters with a text input based on statistical learning, this paper proposes extending this model to a context-dependent one. We further propose a parameter training algorithm for the present model based on a decision tree-based context clustering.

Index Terms: Speech F_0 contours, stochastic model, Fujisaki model, hidden Markov model, EM algorithm

1. Introduction

The fundamental frequency (F_0) contour of speech is a feature that represents the intonation of an utterance. One important challenge of the text-to-speech synthesis research involves developing a reasonable model and method for generating a natural-sounding F_0 contour from a text input.

Thanks to the increasing availability of speech databases, speech synthesis systems based on statistical models such as hidden Markov models (HMMs) have attracted particular attention in recent years. In the HMM-based speech synthesis system [2], a sequence of spectra, F_0s and their delta and acceleration components is modeled simultaneously within a unified HMM framework. At the synthesis stage, a sequence of these parameters is generated according to the output probabilities of the trained HMM given an input sentence. The constraints of the dynamic parameters are considered during parameter generation in order to guarantee the smoothness of the generated spectral and F_0 trajectories. However, conventional HMM based speech synthesis systems tend to produce over-smoothed F_0 contours, which often result in a synthesis that sounds “emotionless” to human listeners.

In speech synthesis technology, one important challenge involves synthesizing an F_0 contour that is not only linguistically appropriate but also physically likely to be generated via the control mechanism of phonation. The Fujisaki model [1] is a well-founded mathematical model, that describes the process by which the whole F_0 contour of a speech utterance is generated. This model is known to approximate actual F_0 contours of speech well when the parameters are chosen appropriately. In the Fujisaki model, F_0 contour on a logarithmic scale is assumed to be the superposition of three components: a phrase component, an accent component and a baseline component. The phrase component consists of the major-scale pitch variations over the duration of the prosodic units, and the accent component consists of the smaller-scale pitch variations in accented syllables. To avoid synthesizing over-smoothed and physically unlikely F_0 contours, one reasonable approach would be to incorporate the Fujisaki model into the statistical model for speech synthesis so that we can separately take the average of each of these components according to the assigned context labels. Thus, the Fujisaki model can potentially be a good model for F_0 contour synthesis. However, since the Fujisaki model does not take the form of a statistical (automatically trainable) model, using the Fujisaki model for synthesizing F_0 contours within a statistical framework is not straightforward. Indeed, estimating (learning) the Fujisaki model parameters from raw F_0 contour observations has been a difficult task. Several techniques have already been developed (e.g., [3, 4, 5, 6]), but so far with limited success due to the difficulty in searching for optimal parameters under the constraints imposed in the Fujisaki model.

We have previously formulated a statistical model of speech F_0 contours by translating the Fujisaki model into a probabilistic model described as a discrete-time stochastic process [7, 8]. This formulation has allowed us not only to derive an efficient parameter inference algorithm utilizing powerful statistical methods but also to obtain an automatically trainable version of the Fujisaki model. The aim of this paper is to further extend this model to a context-dependent one so as to be able to learn and generate the Fujisaki parameters from input sentences.

The rest of this paper is organized as follows. Sec. 2 briefly reviews the original Fujisaki model and a discrete-time stochastic counterpart to the Fujisaki model, that we have introduced in [7, 8]. Sec. 3 proposes to extend it to a context-dependent one. Sec. 4 proposes a parameter training algorithm for the present model based on decision tree-based context clustering. Sec. 5 shows some results of a speech synthesis experiment conducted using real speech data excerpted from the ATR speech database. Sec. 6 concludes this paper.

2. Generative model of speech F_0 contours

2.1. Original Fujisaki Model

The Fujisaki model [1] assumes that an F_0 contour on a logarithmic scale, $y(t)$, where t is time, is the superposition of three components: a phrase component $y_{\rm p}(t)$, an accent component $y_{\rm a}(t)$, and a base component $y_{\rm b}$:

$$y(t) = y_{\rm p}(t) + y_{\rm a}(t) + y_{\rm b}. \tag{1}$$

The phrase component $y_{\rm p}(t)$ consists of the major-scale pitch variations over the duration of the prosodic units, and the accent component $y_{\rm a}(t)$ consists of the smaller-scale pitch variations in accentual phrases. These two components are modeled as the outputs of second-order critically damped filters, one being excited with a command function $u_{\rm p}(t)$ consisting of Dirac deltas (phrase commands), and the other with $u_{\rm a}(t)$ consisting of rectangular pulses (accent commands):

$$y_{\rm p}(t) = G_{\rm p}(t) * u_{\rm p}(t), \tag{2}$$

$$G_{\rm p}(t) = \begin{cases} \alpha^2 t e^{-\alpha t} & (t \geq 0) \\ 0 & (t < 0) \end{cases}, \tag{3}$$

$$y_{\rm a}(t) = G_{\rm a}(t) * u_{\rm a}(t), \tag{4}$$

$$G_{\rm a}(t) = \begin{cases} \beta^2 t e^{-\beta t} & (t \geq 0) \\ 0 & (t < 0) \end{cases}, \tag{5}$$

where $*$ denotes convolution over time. The baseline component $y_{\rm b}$ is a constant value related to the lower bound of the speaker's F_0, below which no regular vocal fold vibration can

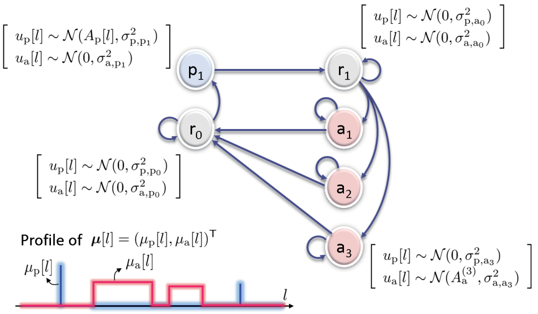

$\left[\begin{array}{l} u_\mathrm{p}[l] \sim \mathcal{N}(A_\mathrm{p}[l], \sigma^2_{\mathrm{p,p_1}}) \\ u_\mathrm{a}[l] \sim \mathcal{N}(0, \sigma^2_{\mathrm{a,p_1}}) \end{array}\right]$
$\left[\begin{array}{l} u_\mathrm{p}[l] \sim \mathcal{N}(0, \sigma^2_{\mathrm{p,a_0}}) \\ u_\mathrm{a}[l] \sim \mathcal{N}(0, \sigma^2_{\mathrm{a,a_0}}) \end{array}\right]$
p1
r1
r0
a1
a2
a3
$\left[\begin{array}{l} u_\mathrm{p}[l] \sim \mathcal{N}(0, \sigma^2_{\mathrm{p,p_0}}) \\ u_\mathrm{a}[l] \sim \mathcal{N}(0, \sigma^2_{\mathrm{a,p_0}}) \end{array}\right]$
Profile of $\boldsymbol{\mu}[l] = (\mu_\mathrm{p}[l], \mu_\mathrm{a}[l])^\mathsf{T}$
$\mu_\mathrm{p}[l]$
$\mu_\mathrm{a}[l]$
l
$\left[\begin{array}{l} u_\mathrm{p}[l] \sim \mathcal{N}(0, \sigma^2_{\mathrm{p,a_3}}) \\ u_\mathrm{a}[l] \sim \mathcal{N}(A^{(3)}_\mathrm{a}, \sigma^2_{\mathrm{a,a_3}}) \end{array}\right]$

Figure 1: *Previous HMM topology for command function modeling.*

be maintained. α and β are natural angular frequencies of the two second-order systems, which are known to be almost constant within an utterance as well as across utterances for a particular speaker. It has been shown that $\alpha = 3$ [rad/s] and $\beta = 20$ [rad/s] can be used as default values.

It is interesting to note that the phrase and accent commands, which we will henceforth refer to as the Fujisaki-model parameters, can be interpreted as quantities related to linguistic information. In the Japanese language, a phrase command and an accent command typically occur at the beginning of each breath group and over the range of accent nucleus in each accentual phrase, respectively.

2.2. Probabilistic formulation of F_0 contour model

Here, we briefly review our probabilistic pitch contour model based on the discrete-time version of the Fujisaki model [7, 8].

In the original Fujisaki model, the phrase commands and accent commands are assumed to consist of Dirac deltas and rectangular pulses, respectively. In addition, they are not allowed to overlap each other. To incorporate these requirements, we find it convenient to model the $u_\mathrm{p}[k]$ and $u_\mathrm{a}[k]$ pair, i.e., $\boldsymbol{o}[k] = (u_\mathrm{p}[k], u_\mathrm{a}[k])^\mathsf{T}$, using a hidden Markov model (HMM). In [7, 8], we have assumed that $\{\boldsymbol{o}[k]\}_{k=1}^K$ is a sequence of outputs generated from an HMM with the specific topology illustrated in Fig. 1. In state r_0, $\mu_\mathrm{p}[k]$ and $\mu_\mathrm{a}[k]$ are both constrained to be zero. In state p_1, referred to as the "phrase state," $\mu_\mathrm{p}[k]$ can take a non-zero value, $A_\mathrm{p}[k]$, whereas $\mu_\mathrm{a}[k]$ is still restricted to zero. At the phrase state, no selftransitions are allowed. In state r_1, $\mu_\mathrm{p}[k]$ and $\mu_\mathrm{a}[k]$ become zero again. This path constraint restricts $\mu_\mathrm{p}[k]$ to consisting of isolated deltas. State r_1 leads to states $\mathrm{a}_1, \ldots, \mathrm{a}_N$, referred to as the "accent states." At each accent state, $\mu_\mathrm{a}[k]$ can take a different non-zero value $A_\mathrm{a}^{(n)}$, whereas $\mu_\mathrm{p}[k]$ is forced to be zero. A direct state transition from an accent state to a different state without passing through state r_1 is not allowed. This path constraint restricts $\mu_\mathrm{a}[k]$ to consisting of rectangular pulses. The output distribution of each state is assumed to be a Gaussian distribution

$$\boldsymbol{o}[k] \sim \mathcal{N}\left(\boldsymbol{o}[k]; \boldsymbol{c}_{s_k}, \boldsymbol{\Upsilon}_{s_k}\right), \tag{6}$$

where s_k indicates the state variable. Namely, the mean vector $\boldsymbol{\mu}[k] = (\mu_\mathrm{p}[k], \mu_\mathrm{a}[k])^\mathsf{T} = \boldsymbol{c}_{s_k}$ and covariance matrix $\boldsymbol{\Sigma}[k] = \boldsymbol{\Upsilon}_{s_k}$ are considered to evolve in time as a result of the state transition $s_1, \ldots, s_K$. The definition of the above HMM can be summarized as follows:

$$\boxed{\begin{array}{rl} \text{Output sequence:} & \{\boldsymbol{o}[k]\}_{k=1}^K \\ \text{State sequence:} & \{s_k\}_{k=1}^K \\ \text{Output distribution:} & P(\boldsymbol{o}[k]|s_k) = \mathcal{N}(\boldsymbol{o}[k]; \boldsymbol{c}_{s_k}, \boldsymbol{\Upsilon}_{s_k}) \\ \text{Mean sequence:} & \boldsymbol{\mu}[k] = (\mu_\mathrm{p}[k], \mu_\mathrm{a}[k])^\mathsf{T} = \boldsymbol{c}_{s_k} \\ \text{Transition probability:} & \phi_{i',i} = \log P(s_k = i | s_{k-1} = i') \end{array}}$$

Given the state sequence $\boldsymbol{s} = \{s_k\}_{k=1}^K$, the above HMM generates the $u_\mathrm{p}[k]$ and $u_\mathrm{a}[k]$ pair. From (2) and (4), $u_\mathrm{p}[k]$ and $u_\mathrm{a}[k]$ are then fed through different critically damped filters, $G_\mathrm{p}[k]$ and $G_\mathrm{a}[k]$, to generate the phrase and accent components, $y_\mathrm{p}[k]$ and $y_\mathrm{a}[k]$:

$$y_\mathrm{p}[k] = u_\mathrm{p}[k] * G_\mathrm{p}[k], \tag{7}$$

$$y_\mathrm{a}[k] = u_\mathrm{a}[k] * G_\mathrm{a}[k], \tag{8}$$

where $*$ denotes convolution over k. An F_0 contour is then given by

$$y[k] = y_\mathrm{p}[k] + y_\mathrm{a}[k] + y_\mathrm{b}, \tag{9}$$

where y_b denotes the baseline value.

3. Context Dependent Generative Model of F_0 Contours

3.1. Context dependent phrase and accent command

As discussed in Sec. 2.1, phrase and accent command is closely associated with linguistic information such as breath group or accent nucleus. Thus we assume that we can obtain an appropriate F_0 contour from a text input by allocating a phrase command to the beggining of each breath group and an accent command to the range of each accent nucleus. Here, the problem is how to determine the magnitudes of the phrase and accent commands. In this paper, we treat the magnitude of each command as a model parameter to be trained using training data. We can assume that commands depend on the types of the preceding, current and succeeding breath groups and accentual phrases because of the fact that F_0 contours represent the intonation of natural speech. Hence, the phrase and accent commands should be determined according to the types of the preceding, current and succeeding breath groups and accentual phrases. We henceforth call those environments context. In Sec. 3, we propose a decision-tree based context clustering algorithm, which allows us to train the decision tree along with the model parameters using the context labels.

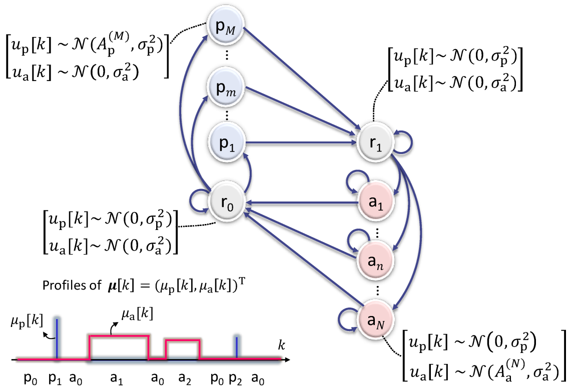
In this paper, the following contextual factors are taken into account:

- Contextual factors that are relevant to accent commands
  - mora count and accent type of {preceeding, current, suceeding} accentual phrase
  - position of current accentual phrase in current breath group
  - position of current accentual phrase in sentence
  - {preceeding, current, suceeding} mora count of breath group
  - position of current breath group in sentence
  - mora count in sentence
- Contextual factors that are relevant to phrase commands
  - {preceeding, current, suceeding} mora count of breath group
  - accentual phrase count in {preceeding, current, suceeding} breath group
  - position of current breath group in sentence
  - mora count in sentence

3.2. Probabilistic F_0 contour model including context dependent phrase and accent command with HMM

To extend the model presented in Subsec. 2.2 to a contex-dependent one, we consider the HMM described in Fig. 2. In this HMM, we would like the phrase and accent states to be grouped into clusters via context clustering. We use M and N to denote the numbers of the phrase states and accent states, respectively. Thus, the present HMM can be summarized as follows:

$$\boxed{\begin{array}{l} \text{Output sequence: } \boldsymbol{o}[k] = (u_\mathrm{p}[k], u_\mathrm{a}[k])^\mathsf{T} \; (k = 1, \ldots, K) \\ \text{Set of states: } \mathcal{S} = \{\mathrm{p}_0, \cdots, \mathrm{p}_M, \mathrm{a}_0, \cdots, \mathrm{a}_N\} \\ \text{State sequence: } \boldsymbol{s} = \{s_k \in \mathcal{S} | k = 1, \ldots, K\} \\ \text{Output distribution: } P(\boldsymbol{o}[k]|s_k = i) = \mathcal{N}(\boldsymbol{c}_i[k], \boldsymbol{\Upsilon}) \\ \boldsymbol{c}_i = \begin{cases} (0,0)^\mathsf{T} & (i \in \mathrm{p}_0, \mathrm{a}_0) \\ (A_\mathrm{p}^{(m)}, 0)^\mathsf{T} & (i \in \mathrm{p}_m) \\ (0, A_\mathrm{a}^{(n)})^\mathsf{T} & (i \in \mathrm{a}_n) \end{cases} \quad \boldsymbol{\Upsilon} = \begin{bmatrix} \upsilon^2_{\mathrm{p},i} & 0 \\ 0 & \upsilon^2_{\mathrm{a,i}} \end{bmatrix} \\ \text{Mean sequence: } \; \boldsymbol{\mu}[k] = (\mu_\mathrm{p}[k], \mu_\mathrm{a}[k])^\mathsf{T} = \boldsymbol{c}_{s_k} \end{array}}$$



$\begin{bmatrix} u_\mathrm{p}[k] \sim \mathcal{N}(A_\mathrm{p}^{(M)}, \sigma_\mathrm{p}^2) \\ u_\mathrm{a}[k] \sim \mathcal{N}(0, \sigma_\mathrm{a}^2) \end{bmatrix}$
p_M
p_m
p_1
r_0
r_1
a_1
a_n
a_N
$\begin{bmatrix} u_\mathrm{p}[k] \sim \mathcal{N}(0, \sigma_\mathrm{p}^2) \\ u_\mathrm{a}[k] \sim \mathcal{N}(0, \sigma_\mathrm{a}^2) \end{bmatrix}$
$\begin{bmatrix} u_\mathrm{p}[k] \sim \mathcal{N}(0, \sigma_\mathrm{p}^2) \\ u_\mathrm{a}[k] \sim \mathcal{N}(0, \sigma_\mathrm{a}^2) \end{bmatrix}$
Profiles of $\boldsymbol{\mu}[k] = (\mu_\mathrm{p}[k], \mu_\mathrm{a}[k])^\mathsf{T}$
$\mu_\mathrm{p}[k]$
$\mu_\mathrm{a}[k]$
k
p_0 p_1 a_0 a_1 a_0 a_2 p_0 p_2 a_0
$\begin{bmatrix} u_\mathrm{p}[k] \sim \mathcal{N}(0, \sigma_\mathrm{p}^2) \\ u_\mathrm{a}[k] \sim \mathcal{N}(A_\mathrm{a}^{(N)}, \sigma_\mathrm{a}^2) \end{bmatrix}$

Figure 2: *Proposed command function generative HMM. Unlike in the case of previous HMM, command function is output from HMM in which states are devided by context clustering.*

At the training stage, the positions of the breath groups and the accent nucleus are assumed to be specified according to the hand-annotated context labels. Namely, the state sequence $\boldsymbol{s}$ is assumed to be given and fixed during the training phase. Now, let us define

$$\boldsymbol{y} = \{y[k]\}_{k=1}^K, \quad \boldsymbol{s} = \{s_k\}_{k=1}^K,$$
$$\boldsymbol{o} = \{(u_\mathrm{p}[k], u_\mathrm{a}[k])^\mathsf{T}\}_{k=1}^K, \quad \boldsymbol{\theta} = \{\{A_\mathrm{p}^{(m)}\}_{m=1}^M, \{A_\mathrm{a}^{(n)}\}_{n=1}^N\},$$
$$\boldsymbol{\mu}_\mathrm{p} = \{\boldsymbol{\mu}_\mathrm{p}[k]\}_{k=1}^K, \quad \boldsymbol{\mu}_\mathrm{a} = \{\boldsymbol{\mu}_\mathrm{a}[k]\}_{k=1}^K.$$

For simplicity, we treat μ_b, σ_p^2, σ_a^2, σ_b^2, α, β as constants. Here, $\boldsymbol{\theta} = \{\{A_\mathrm{p}^{(m)}\}_{m=1}^M, \{A_\mathrm{a}^{(n)}\}_{n=1}^N\}$ are the free parameters to be trained. In the same way as [7], the likelihood function of the Fujisaki model parameters $\boldsymbol{\theta}$ given $\boldsymbol{y}$ can be derived as

$$P(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{|\boldsymbol{\Sigma}^{-1}|^{1/2}}{(2\pi)^{K/2}} \exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})^\mathsf{T}\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})\right\},$$
$$\boldsymbol{\mu} = \boldsymbol{A}^{-1}\boldsymbol{\mu}_\mathrm{p} + \boldsymbol{B}^{-1}\boldsymbol{\mu}_\mathrm{a} + \mu_\mathrm{b}\boldsymbol{1}, \tag{10}$$
$$\boldsymbol{\Sigma} = \boldsymbol{A}^{-1}\boldsymbol{\Sigma}_\mathrm{p}\left(\boldsymbol{A}^\mathsf{T}\right)^{-1} + \boldsymbol{B}^{-1}\boldsymbol{\Sigma}_\mathrm{a}\left(\boldsymbol{B}^\mathsf{T}\right)^{-1} + \boldsymbol{\Sigma}_\mathrm{b},$$

where

$$\boldsymbol{\Sigma}_\mathrm{p} = \sigma_\mathrm{p}^2\boldsymbol{I}, \ \boldsymbol{\Sigma}_\mathrm{a} = \sigma_\mathrm{a}^2\boldsymbol{I}, \ \boldsymbol{\Sigma}_\mathrm{b} = \sigma_\mathrm{b}^2\boldsymbol{I},$$

$$\boldsymbol{A} = \begin{bmatrix} a_0 & & & & O \\ a_1 & a_0 & & & \\ a_2 & a_1 & a_0 & & \\ & \ddots & \ddots & \ddots & \\ O & & a_2 & a_1 & a_0 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} b_0 & & & & O \\ b_1 & b_0 & & & \\ b_2 & b_1 & b_0 & & \\ & \ddots & \ddots & \ddots & \\ O & & b_2 & b_1 & b_0 \end{bmatrix},$$

$$a_2 = (\psi-1)^2, a_1 = -2\psi(\psi-1), a_0 = \psi^2,$$
$$b_2 = (\varphi-1)^2, b_1 = -2\varphi(\varphi-1), b_0 = \varphi^2,$$
$$\psi = 1 + \frac{1}{\alpha t_0}, \varphi = 1 + \frac{1}{\beta t_0}.$$

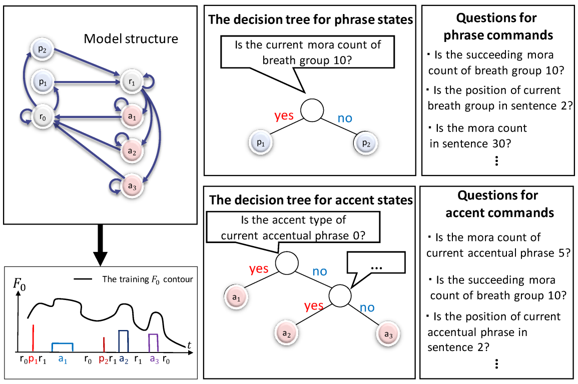
4. Parameter Training and Generation Processes

4.1. Context clustering

In this section we propose an algorithm for training model parameters $\boldsymbol{\theta} = \{\{A_\mathrm{p}^{(m)}\}_{m=1}^M, \{A_\mathrm{a}^{(n)}\}_{n=1}^N\}$ based on decision tree context clustering [11] and the expectation-maximization (EM) algorithm [7, 8]. This algorithm allows us to train model parameters using the training data and to generate phrase and command functions from an input sentence. In the following, we will describe the case where the minimum description length (MDL) criterion is used for selecting nodes to be split. We also select the probability density function of Fujisaki model parameters $\boldsymbol{\theta}$ and state sequence $\boldsymbol{s}$ given F_0 contours $\boldsymbol{y}$ as the likelihood in the MDL criterion. Here, the number of the leaf nodes of the decision tree about phrase commands is equal to the number of the phrase states p_m, i.e., M. The number of the leaf nodes of the decision tree about accent commands is also equal to the number of the accent states a_n in the present HMM, i.e., N. Let us define $d = 1, \ldots, D$ as the index of the sentence in the training data, the F_0 contour of d-th sentence as $\boldsymbol{y}^{(d)} = \{y^{(d)}[k]\}_{k=1}^{K^{(d)}}$ and $\boldsymbol{\theta}$ as the model parameters. Then the MDL is given by

$$MDL = -L(\boldsymbol{\theta}) + c(N+M)\log W + C,$$
$$L(\boldsymbol{\theta}) = \sum_{d=1}^{D}\left\{\frac{1}{2}\log|\boldsymbol{\Sigma}^{-1}| - \frac{K^{(d)}}{2}\log 2\pi \right.$$
$$\left. - \frac{1}{2}(\boldsymbol{y}^{(d)}-\boldsymbol{\mu}^{(d)})^\mathsf{T}\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}^{(d)}-\boldsymbol{\mu}^{(d)})\right\}, \tag{11}$$
$$\boldsymbol{\mu}^{(d)} = \boldsymbol{A}^{-1}\boldsymbol{\mu}_\mathrm{p}^{(d)} + \boldsymbol{B}^{-1}\boldsymbol{\mu}_\mathrm{a}^{(d)} + \mu_\mathrm{b}\boldsymbol{1},$$
$$\boldsymbol{\Sigma} = \boldsymbol{A}^{-1}\boldsymbol{\Sigma}_\mathrm{p}\left(\boldsymbol{A}^\mathsf{T}\right)^{-1} + \boldsymbol{B}^{-1}\boldsymbol{\Sigma}_\mathrm{a}\left(\boldsymbol{B}^\mathsf{T}\right)^{-1} + \boldsymbol{\Sigma}_\mathrm{b},$$

where $L(\boldsymbol{\theta})$ denotes the log-likelihood function defined by (10),c denotes the weighting factor for the adjusting model size, and C denotes the code length required for choosing the model, which is assumed to be constant. Now we have to reestimate the model parameter $\boldsymbol{\theta}$ each time a node is split during the context clustering process. An efficient parameter estimation algorithm under a fixed model structure has already been proposed in [7]. Note that the proposed method is different from the previous one in that the state sequence $\boldsymbol{s}$ is fixed and that the model parameters $\boldsymbol{\theta}$ is defined differently. The overview of the proposed method is shown in Fig. 3



Model structure
p_2
p_1
r_0
r_1
a_1
a_2
a_3
The decision tree for phrase states
Is the current mora count of breath group 10?
yes
no
p_1
p_2
Questions for phrase commands
• Is the succeeding mora count of breath group 10?
• Is the position of current breath group in sentence 2?
• Is the mora count in sentence 30?
⋮
F_0
The training F_0 contour
t
$\mathrm{r}_0 \mathrm{p}_1 \mathrm{r}_1$ a_1 r_0 $\mathrm{p}_2 \mathrm{r}_1$ a_2 r_1 a_3 r_0
The decision tree for accent states
Is the accent type of current accentual phrase 0?
yes
no
…
yes
no
a_1
a_2
a_3
Questions for accent commands
• Is the mora count of current accentual phrase 5?
• Is the succeeding mora count of breath group 10?
• Is the position of current accentual phrase in sentence 2?
⋮

Figure 3: *Overview of the proposed context clustering algorithm.*

4.2. Parameter training algorithm

Here we present a parameter inference algorithm that searches for the unknown model parameter $\boldsymbol{\theta}$ by iteratively updating $\boldsymbol{\theta}$ so as to maximize a lower bound function of the log-likelihood $\sum_d \log P(\boldsymbol{y}^{(d)}|\boldsymbol{\theta})$. For simplicity, we will hereafter omit the superscript d in $\boldsymbol{y}^{(d)}$. By regarding $\boldsymbol{x} = (\boldsymbol{y}_\mathrm{p}^\mathsf{T}, \boldsymbol{y}_\mathrm{a}^\mathsf{T}, \boldsymbol{y}_\mathrm{b}^\mathsf{T})^\mathsf{T}$ as the complete data this problem can be viewed as an incomplete data problem, which can be dealt with using the EM algorithm. The log-likelihood function of $\boldsymbol{\theta}$ given $\boldsymbol{x}$ is written as

$$\log P(\boldsymbol{x}|\boldsymbol{\theta}) \overset{c}{=} \frac{1}{2}\log|\boldsymbol{\Lambda}^{-1}| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}(\boldsymbol{x}-\boldsymbol{m}),$$
$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{y}_{\mathrm{p}}\\ \boldsymbol{y}_{\mathrm{a}}\\ \boldsymbol{y}_{\mathrm{b}}\end{bmatrix},\ \boldsymbol{m} = \begin{bmatrix}\boldsymbol{A}^{-1}\boldsymbol{\mu}_{\mathrm{p}}\\ \boldsymbol{B}^{-1}\boldsymbol{\mu}_{\mathrm{a}}\\ \mu_{\mathrm{b}}\boldsymbol{1}\end{bmatrix}, \qquad (12)$$
$$\boldsymbol{\Lambda}^{-1} = \begin{bmatrix}\boldsymbol{A}^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathrm{p}}^{-1}\boldsymbol{A} & \boldsymbol{O} & \boldsymbol{O}\\ \boldsymbol{O} & \boldsymbol{B}^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathrm{a}}^{-1}\boldsymbol{B} & \boldsymbol{O}\\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{\Sigma}_{\mathrm{b}}^{-1}\end{bmatrix}.$$

In this case the Q function is thus given by

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}') \overset{c}{=} \frac{1}{2}\Big[\log|\boldsymbol{\Lambda}^{-1}| - \mathrm{tr}(\boldsymbol{\Lambda}^{-1}\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^{\mathsf{T}}|\boldsymbol{y};\boldsymbol{\theta}'])$$
$$+ 2\boldsymbol{m}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\theta}'] - \boldsymbol{m}^{\mathsf{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{m}\Big]. \qquad (13)$$

In above equation, a prior probability $\Pr(\boldsymbol{\theta})$ is constant since $\boldsymbol{\theta}$ is uniformity distributed and the state sequence $\boldsymbol{s}$ is fixed.

$\mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\theta]$ and $\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^{\mathsf{T}}|\boldsymbol{y};\boldsymbol{\theta}]$ are given explicitly as

$$\mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\theta] = \boldsymbol{m} + \boldsymbol{\Lambda}\boldsymbol{H}^{\mathsf{T}}(\boldsymbol{H}\boldsymbol{\Lambda}\boldsymbol{H}^{\mathsf{T}})^{-1}(\boldsymbol{y}-\boldsymbol{H}\boldsymbol{m}),$$
$$\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^{\mathsf{T}}|\boldsymbol{y};\theta] = \boldsymbol{\Lambda} - \boldsymbol{\Lambda}\boldsymbol{H}^{\mathsf{T}}(\boldsymbol{H}\boldsymbol{\Lambda}\boldsymbol{H}^{\mathsf{T}})^{-1}\boldsymbol{H}\boldsymbol{\Lambda}$$
$$+ \mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\theta]\mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\theta]^{\mathsf{T}},$$

by using the relationship $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x}$, where $\boldsymbol{H} = [\boldsymbol{I},\boldsymbol{I},\boldsymbol{I}]$. These are the values to be updated at the E step. Let $\mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\theta]$ be partitioned into four $K \times 1$ blocks such that $\mathbb{E}[\boldsymbol{x}|\boldsymbol{y};\theta] = (\bar{\boldsymbol{x}}_{\mathrm{p}}^{\mathsf{T}}, \bar{\boldsymbol{x}}_{\mathrm{a}}^{\mathsf{T}}, \bar{\boldsymbol{x}}_{\mathrm{b}}^{\mathsf{T}})^{\mathsf{T}}$.

In the M step, There is no need to update the state sequence parameter $\boldsymbol{s}$ since we set restriction that it is always fixed. Hence the M step update is equivalent to the procedure that maximize with respect to $A_{\mathrm{p}}^{(m)}$ and $A_{\mathrm{a}}^{(n)}$ and is as follows.

$$A_{\mathrm{p}}^{(m)} = \frac{1}{|\mathcal{T}_{\mathrm{p}_m}|}\sum_{k\in\mathcal{T}_{\mathrm{p}_m}}[\boldsymbol{A}\bar{\boldsymbol{x}}_{\mathrm{p}}]_k,\ \ \mathcal{T}_{\mathrm{p}_m} = \{k|s_k = \mathrm{p}_m\},$$
$$A_{\mathrm{a}}^{(n)} = \frac{1}{|\mathcal{T}_{\mathrm{a}_n}|}\sum_{k\in\mathcal{T}_{\mathrm{a}_n}}[\boldsymbol{B}\bar{\boldsymbol{x}}_{\mathrm{a}}]_k,\ \ \mathcal{T}_{\mathrm{a}_n} = \{k|s_k = \mathrm{a}_n\}.$$

4.3. Parameter generation process

To obtain F_0 contour from input text, we firstly extract the state sequence by using context and constructed decision trees by training. Then, according to state sequence $\boldsymbol{s}$ and model parameters $\boldsymbol{\theta}$, we can obtain the F_0 contour by determining $\boldsymbol{y}$ so as to maximize $P(\boldsymbol{y}|\boldsymbol{s},\boldsymbol{\theta})$ with respect to $\boldsymbol{y}$.

5. Preliminary experiment for proposed method

5.1. Experimental conditions

As a preliminary experiment, we implemented a simplified version of the present method to demonstrate the proof of concept of the present method. The simplified version we have implemented consists of two stages: First, we extract phrase and accent commands using the method described in [8] from the raw F_0 contours of the training data (the first 450 sentences of HTS ver 2.1 demo script [9]). Second, the values of the extracted phrase and accent commands were clustered using a decision tree based context clustering. The decision tree was constructed according to the following criterion:

$$MDL = \frac{1}{2}\sum_{d=1}^{D_j}\left\{\log(2\pi\sigma_j^2) + \frac{(x_i-\mu_j)}{\sigma_j^2}\right\} + cJ\log W \qquad (14)$$

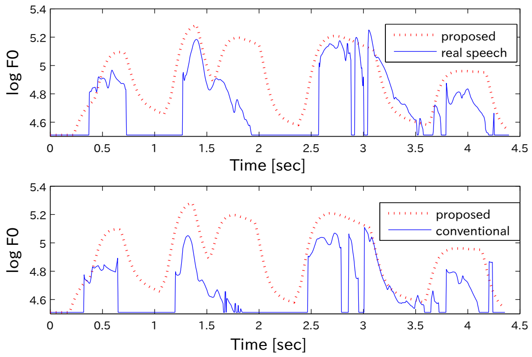


Figure 4: *An example of generated F_0 contours of a japanese utterance "ippyo no kakusa wa sarani hirogaru darou". The top graph shows the F_0 contour generated by the simplified version of the present method described in 5.1 along with the F_0 contour of real speech. The bottom graph shows the F_0 contour generated by HTS [2].*

where x_i indecates the magnitude of each command, J indicates the number of nodes in each tree, j indicates node index, and D_j indicates the state occupancy of j-th node. We used the HTS label sequence in [9] in order to extract the beginning of each breath group and the range of each accentual phrase. In the method [8], the constant parameters were fixed respectively at $t_0 = 8$ ms, $\alpha = 3.0$ rad/s, $\beta = 20.0$ rad/s, $v_{\mathrm{n}}^2[k] = 10^{15}$ for unvoiced regions and $v_{\mathrm{n}}^2[k] = 0.2^2$ for voiced regions. μ_{b} was set at the minimum $\log F_0$ value in the voiced regions.

5.2. Experimental results

Fig. 4 shows an example of the F_0 contours generated using the last 53 sentences in the speech database available at [9], the raw F_0 contours of real speech extracted with the straight analysis [10] and the F_0 contours generated by HTS [2]. Since the conventional method is not ensured to generate an F_0 contour that follows the generating process of F_0 contours, it tended to sometimes generate unnatural-sounding F_0 contours. On the other hand, most of the F_0 contours generated by the present method sounded reasonably natural. This may be due to the fact that the Fujisaki model is able to express human F_0 contours consistently well. Some examples of the synthesized speech generated by the present and conventional methods are demonstrated in our demo site {`http://hil.t.u-tokyo.ac.jp/~kadowaki/Demos.htm`}. We can also confirm from these results that the over-smoothing of F_0 contours rarely occurred with the present approach.

6. Conclusion

This paper proposed a method for generating the F_0 contour of speech from a text input for text-to-speech synthesis. We previously introduced a statistical model describing the generating process of speech F_0 contours, based on the discrete-time version of the Fujisaki model. To associate a sequence of the Fujisaki-model parameters with a text input based on statistical learning, we extended this model to a context-dependent one. This idea was motivated by our expectation that averaging these parameters would not directly cause the over-smoothing of the F_0 contours, unlike the conventional method. We further proposed a parameter training algorithm for the present model based on a decision tree-based context clustering. The preliminary experimental results revealed that the present method was able to generate natural-sounding F_0 contours. Future work includes implementing the unified training algorithm described in Sec. 4 and conducting subjective evaluations.

7. acknowledgement

This work was supported by JSPS KAKENHI Grant Number 26280060.

8. References

[1] H. Fujisaki, *In Vocal Physiology: Voice Production, Mechanisms and Functions*, Raven Press, 1988.

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

[3] H. Fujisaki and K. Hirose, "Analysis of voice fundamental frequency contours for declarative sentences of japanese," *J. Acoust. Soc. Jpn (E)*, vol. 5, no. 4, pp. 233–242, 1984.

[4] H. Mixdorf, "A novel approach to the fully automatic extraction of fujisaki model parameters," in *Proc. ICASSP*, 2000, vol. 3, pp. 1281–1284.

[5] S. Narusawa, N. Minematsu, K. Hirose, and H. Fujisaki, "A method for automatic extraction of model parameters from fundamental frequency contours of speech," in *Proc. ICASSP*, 2002, pp. 509–512.

[6] Xu, Yi, and Santitham Prom-On, "Toward invariant functional representations of variable surface fundamental frequency contours: Synthesizing speech melody via model-based stochastic learning," in *Speech Communication 57*, 2014, pp. 181–208.

[7] H. Kameoka, J. L. Roux, and Y. Ohishi, "A statistical model of speech F_0 contours," in *Proc. SAPA*, 2010, pp. 43–48.

[8] K. Yoshizato, H. Kameoka, D. Saito, and S. Sagayama, "Statistical approach to fujisaki-model parameter estimation from speech signals and its quantitative evaluation," in *Proc. Speech Prosody 2012*, 2012, pp. 175–178.

[9] "HMM-based Speech Synthesis System (HTS)," `http://hts.sp.nitech.ac.jp/`

[10] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F_0 extraction: Possible role of a repetitive structure in sounds," Speech Communication, vol. 27, no. 3, pp. 187–207, 1999.

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