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 natural 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, and the accent component consists of the smaller-scale pitch variations over the duration of the prosodic units, and the phrase component consists of the major-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 oversmoothed 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 the 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 $y_p(t)$, an accent component $y_a(t)$, and a base component $y_b(t)$:

$$\begin{align*}
y(t) = y_p(t) + y_a(t) + y_b(t).
\end{align*}$$

The phrase component $y_p(t)$ consists of the major-scale pitch variations over the duration of the prosodic units, and the accent component $y_a(t)$ consists of the smaller-scale pitch variations in accented syllables. These two components are modeled as the outputs of second-order critically damped filters, one being excited with a command function $u_p(t)$ consisting of Dirac deltas (phrase commands), and the other with $u_a(t)$ consisting of rectangular pulses (accent commands):
where * denotes convolution over time. The baseline component $y_0$ is a constant value related to the lower bound of the speaker’s $F_0$, below which no regular vocal fold vibration can be maintained. $\alpha$ and $\beta$ 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 first consider the $u_p[k]$ and $u_a[k]$ pair, i.e., $o[k] = (u_p[k], u_a[k])^T$, using a hidden Markov model (HMM). In [7, 8], we have assumed that $\{o[k]\}_{k=1}^n$ is a sequence of outputs generated from an HMM with the topology illustrated in Fig. 1. In state $r_0$, $\mu_p[k]$ and $\mu_a[k]$ are both constrained to be zero. In state $r_1$, referred to as the “phrase state,” $\mu_p[k]$ can take a non-zero value, $A_p[k]$, whereas $\mu_a[k]$ is still restricted to zero. At the phrase state, no self-transitions are allowed. In state $r_1$, $\mu_p[k]$ and $\mu_a[k]$ become zero again. This path constraint restricts $\mu_p[k]$ to consisting of isolated deltas. State $r_1$ leads to states $s_1, \ldots, s_K$, referred to as the “accent states.” At each accent state, $\mu_a[k]$ can take a different non-zero value $A_a[k]$, whereas $\mu_p[k]$ 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 $\mu_a[k]$ to consisting of rectangular pulses. The output distribution of each state is assumed to be a Gaussian distribution

$$o[k] \sim \mathcal{N}(o[k]; \mu_s, \Sigma_s),$$

where $s_k$ indicates the state variable. Namely, the mean vector $\mu[k] = (\mu_p[k], \mu_a[k])^T = \mu_s$ and covariance matrix $\Sigma[k] = \Sigma_s$ are considered to evolve in time as a result of the state transition $s_k \rightarrow s_k$. The definition of the above HMM can be summarized as follows:

- **Output sequence:** $\{o[k]\}_{k=1}^K$
- **State sequence:** $\{s_k\}_{k=1}^K$
- **Output distribution:** $P(o[k]|s_k) = \mathcal{N}(o[k]; \mu_s, \Sigma_s)$
- **Mean sequence:** $\mu[k] = (\mu_p[k], \mu_a[k])^T = \mu_s$
- **Transition probability:** $\gamma_{i\rightarrow j} = \log P(s_k = i|s_{k-1} = j)$

Given the state sequence $s = \{s_k\}_{k=1}^K$, the above HMM generates the $u_p[k]$ and $u_a[k]$ pair. From (2) and (4), $u_p[k]$ and $u_a[k]$ are then fed through different critically damped fil-

### 3. Context Dependent Generating 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 beginning 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 these 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 {preceding, current, succeeding} accentual phrase
  - position of current accentual phrase in current breath group
  - position of current accentual phrase in sentence
  - {preceding, current, succeeding} mora count of breath group
  - position of current breath group in sentence
  - mora count in sentence

- **Contextual factors that are relevant to phrase commands**
  - {preceding, current, succeeding} mora count of breath group
  - accentual phrase count in {preceding, current, succeeding} 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 context-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 $\hat{M}$ and $\hat{N}$ to denote the numbers of the phrase states and accent states, respectively. Thus, the present HMM can be summarized as follows: 

$$y_1[k] = u_p[k] * G_p[k],$$

$$y_2[k] = u_a[k] * G_a[k],$$

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

$$y[k] = y_1[k] + y_2[k] + y_0,$$

where $y_0$ denotes the baseline value.
Figure 2: Proposed command function generative HMM. Unlike in the case of previous HMM, command function is output from HMM in which states are decided 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 $s$ is assumed to be given and fixed during the training phase. Now, let us define

$y = \{y[k]\}_{k=1}^{K}, \quad s = \{s[k]\}_{k=1}^{K}, \quad o = \{(u_p[k], u_a[k])\}_{k=1}^{K}, \quad \theta = \{\{A_p^{(m)}\}_{m=1}^{M}, \{A_a^{(n)}\}_{n=1}^{N}\}, \quad \mu_y = \{\mu_p[k], \mu_a[k]\}_{k=1}^{K}$.

For simplicity, we treat $\mu_0, \sigma_p^2, \sigma_a^2, \alpha, \beta$ as constants.

Here, $\theta = \{\{A_p^{(m)}\}_{m=1}^{M}, \{A_a^{(n)}\}_{n=1}^{N}\}$ are the free parameters to be trained. In the same way as in [7], the likelihood function of the Fujisaki model parameters $\theta$ given $y$ can be derived as

$P(y|\theta) = \frac{\Sigma_{p}^{-1/2}}{(2\pi)^{K/2}} \exp\left\{-\frac{1}{2}(y - \mu)^T \Sigma_{p}^{-1}(y - \mu)\right\},$

$\mu = A^{-1} \mu_p + B^{-1} \mu_a + \mu_1,$

$\Sigma = A^{-1} \Sigma_p (A^T)^{-1} + B^{-1} \Sigma_a (B^T)^{-1} + \Sigma_0,$

where

$\Sigma_p = \sigma_p^2 I, \Sigma_a = \sigma_a^2 I, \Sigma_0 = \sigma_0^2 I,$

$\begin{align*}
A &= \begin{bmatrix}
\alpha_0 & \alpha_2 & \alpha_3 & \ldots \\
\alpha_2 & a_1 & a_0 & \ldots \\
\ldots & \ldots & \ldots & \\
\alpha_3 & \alpha_2 & a_1 & a_0 \\
\end{bmatrix}, \\
\mu &= \begin{bmatrix}
\mu_0 \\
\mu_2 \\
\ldots \\
\mu_3 \\
\end{bmatrix}, \\
B &= \begin{bmatrix}
b_0 & b_2 & b_1 & b_0 \\
\ldots & \ldots & \ldots & \\
b_3 & b_2 & b_1 & b_0 \\
\end{bmatrix}, \\
\Sigma &= \begin{bmatrix}
\Sigma_p & 0 & 0 & \ldots \\
0 & \Sigma_a & 0 & \ldots \\
0 & 0 & \Sigma_0 & \ldots \\
0 & \ldots & \ldots & \Sigma_0 \\
\end{bmatrix}.
\end{align*}$

4. Parameter Training and Generation Processes

4.1. Context clustering

In this section we propose an algorithm for training model parameters $\theta = \{\{A_p^{(m)}\}_{m=1}^{M}, \{A_a^{(n)}\}_{n=1}^{N}\}$ 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 $\theta$ and state sequence $s$ given $F_0$ contours $y$ 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 $y^{(d)} = \{y^{(d)}[k]\}_{k=1}^{K}$, and $\theta$ as the model parameters. Then the MDL is given by

$MDL = -L(\theta) + c(N + M) \log W + C,$

$L(\theta) = \sum_{d=1}^{D} \left\{ \frac{1}{2} \log |\Sigma| - \frac{1}{2} \log 2\pi \\
- \frac{1}{2} (y^{(d)} - \mu^{(d)})^T \Sigma^{-1} (y^{(d)} - \mu^{(d)}) \right\},$

$\mu^{(d)} = A^{-1} \mu_p^{(d)} + B^{-1} \mu_a^{(d)} + \mu_1,$

$\Sigma = A^{-1} \Sigma_p (A^T)^{-1} + B^{-1} \Sigma_a (B^T)^{-1} + \Sigma_0,$

where $L(\theta)$ 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 $\theta$ 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 $s$ is fixed and that the model parameters $\theta$ is defined differently. The overview of the proposed method is shown in Fig. 3.

4.2. Parameter training algorithm

Here we present a parameter inference algorithm that searches for the unknown model parameter $\theta$ by iteratively updating $\theta$ so as to maximize a lower bound function of the log-likelihood $\sum_{d=1}^{D} \log P(y^{(d)}|\theta)$. For simplicity, we will hereafter omit the superscript $d$ in $y^{(d)}$. By regarding $x = (y_0^T, y_0^T, y_0^T)$ 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 $\theta$ given $x$ is written as

$$
\log P(x; \theta) = \frac{1}{2} \log |A| - \frac{1}{2} (x - m)^T A^{-1} (x - m),
$$

$$
x = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad m = \begin{bmatrix} A^{-1} \mu_x \\ B^{-1} \mu_y \end{bmatrix},
$$

$$
A^{-1} = \begin{bmatrix} A^{-1} \Sigma_y^{-1} A & O \\ O & B^{-1} \Sigma_x^{-1} B \end{bmatrix}.
$$

In this case the Q function is thus given by

$$
Q(\theta, \theta') = \frac{1}{2} \left[ \log |A| - \log |A|' - 2 \log |\Lambda| + 2 \log |\Lambda'| + \alpha \sum_{a \in \Lambda} \log \frac{|\Lambda_a|}{|\Lambda_a'|} \right] + 2 \sum_{s \in \Lambda} \alpha_s (\theta' - \theta) + m^T A^{-1} m - m'^T A'^{-1} m'.
$$

In above equation, a prior probability $P_r(\theta)$ is constant since $\theta$ is uniformly distributed and the state sequence $s$ is fixed.

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

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

by using the relationship $y = Hx$, where $H = [I, I, I]$. These are the values to be updated at the E step. Let $E_x[y|\theta]$ be partitioned into four $K \times 1$ blocks such that $E_x[y|\theta] = (x_j, x_k, x_m)^T$.

In the M step, there is no need to update the state sequence parameter $s$ since we set restriction that it is always fixed. Hence the M step update is equivalent to the procedure that maximize with respect to $A_p(m)$ and $A_a(n)$ and is as follows.

$$
A_p(m) = \frac{1}{|T_p_m|} \sum_{k \in T_p_m} |A_x|^T x_k |, \quad T_p_m = \{ k | s_k = p_m \},
$$

$$
A_a(n) = \frac{1}{|T_a_n|} \sum_{k \in T_a_n} |B_x|^T x_k |, \quad T_a_n = \{ k | s_k = a_n \}.
$$

4.3. Parameter generation process

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

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:

$$
M DL = \frac{1}{2} \sum_{k=1}^{D_t} \left( \log (2 \pi \sigma_j^2) + \frac{(x_{j_i} - \mu_j)^2}{\sigma_j^2} \right) + c J \log W
$$

5.2. Experimental results

Fig. 4 shows 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].

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. References