Speech Synthesis

Basics of HMM-based speech synthesis

Advanced Signal Processing 2, Seminar
Summer term 2008

Written by
Andrea Sereinig

Graz, 2.5.2008

Contents

1 Speech Parameter Generation Using Dynamic Features [1]

1.1 Problem

Let $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T\}$ be the vector sequence of a speech parameter and $\mathbf{q} = \{q_1, q_2, \ldots, q_T\}$ be the state sequence of an HMM λ.

$\mathbf{o}_t = \{\mathbf{c}_t, \Delta\mathbf{c}_t\}$ shall be the vector of a speech parameter at time t where *ct* is the static feature vector (e.g. cepstral coefficients) and *Δct* is the dynamic feature vector (e.g. delta cepstral coefficients).
The goal of the algorithm proposed here is to determine the parameter sequence **c** that maximizes

$$P[\mathbf{O} \mid \lambda] = \sum_{\text{all } \mathbf{q}} P[\mathbf{q}, \mathbf{O} \mid \lambda]$$

1.2 Solution to the Problem

A solution to the problem given above is proposed in [1].
First we state, that for a given **q**, maximizing $P[\mathbf{q}, \mathbf{O} \mid \lambda]$ with respect to **c** is equivalent to maximizing $P[\mathbf{O} \mid \mathbf{q}, \lambda]$ with respect to **c**, since $P[\mathbf{q} \mid \lambda]$ does not depend on **O**.
The probability of making an observation **O** given **q** and λ can be rewritten by $P[\mathbf{O} \mid \mathbf{q}, \lambda] = b_{q_1}(\mathbf{o}_1)\, b_{q_2}(\mathbf{o}_2) \cdots b_{q_T}(\mathbf{o}_T)$ where $\{b_{q_t}(\mathbf{o}_t)\}$ are single mixture Gaussian distributions.
Now, by setting $\partial \log P[\mathbf{O} \mid \mathbf{q}, \lambda] / \partial \mathbf{c} = \mathbf{0}_{TM}$ to maximize $\log P[\mathbf{O} \mid \mathbf{q}, \lambda$ we obtain a set of equations

$$\mathbf{R}\mathbf{c} = \mathbf{r}$$

with

$$\mathbf{R} = \mathbf{U}^{-1} + \mathbf{W}'\Delta\mathbf{U}^{-1}\mathbf{W}$$
$$\mathbf{r} = \mathbf{U}^{-1}\boldsymbol{\mu} + \mathbf{W}'\Delta\mathbf{U}^{-1}\Delta\boldsymbol{\mu}$$

where $\boldsymbol{\mu} = [\boldsymbol{\mu}'_{q_1}, \boldsymbol{\mu}'_{q_2}, \ldots, \boldsymbol{\mu}'_{q_T}]'$ and $\Delta\boldsymbol{\mu} = [\Delta\boldsymbol{\mu}'_{q_1}, \Delta\boldsymbol{\mu}'_{q_2}, \ldots, \Delta\boldsymbol{\mu}'_{q_T}]'$ are the mean vectors of *ct* and *Δc*, respectively and $\mathbf{U} = \mathrm{diag}[\mathbf{U}_{q_1}, \mathbf{U}_{q_2}, \ldots, \mathbf{U}_{q_T}]$ and $\Delta\mathbf{U} = \mathrm{diag}[\Delta\mathbf{U}_{q_1}, \Delta\mathbf{U}_{q_2}, \ldots, \Delta\mathbf{U}_{q_T}]$ are the covariance matrices of *ct* and *Δct*, respectively.

To now obtain optimal values for **q** and **c** (i.e. those which maximize $P[\mathbf{q}, \mathbf{O} \mid \lambda]$), the set of equations $\mathbf{Rc} = \mathbf{r}$ has to be solved for every possible state sequence.

The overall procedure for parameter generation can be summarized as follows:

1. Solve the set of equations $\mathbf{Rc} = \mathbf{r}$ for an initial state sequence and obtain **c** and **P** (use Viterbi algorithm)
2. Replace the state q_t of a frame t with q'_t according to a certain strategy and obtain **c'** and **P'** by using the algorithm in Table 1.
3. If the value of $\log P[\hat{\mathbf{q}}, \hat{\mathbf{O}} \mid \lambda]$ is smaller than that of $\log P[\mathbf{q}, \mathbf{O} \mid \lambda]$, discard the replacement
4. Repeat 2 and 3 until a certain condition is satisfied.

- Set $\mathbf{D}$, $\mathbf{d}$, $\mathbf{w}$ by (23)–(25) to replace $\{\Delta\boldsymbol{\mu}_{q_t}, \Delta\mathbf{U}_{q_t}\}$ with $\{\Delta\boldsymbol{\mu}_{\hat{q}_t}, \Delta\mathbf{U}_{\hat{q}_t}\}$.
- Set $\mathbf{D}$, $\mathbf{d}$, $\mathbf{w}$ by (26)–(28) to replace $\{\boldsymbol{\mu}_{q_t}, \mathbf{U}_{q_t}\}$ with $\{\boldsymbol{\mu}_{\hat{q}_t}, \mathbf{U}_{\hat{q}_t}\}$.

Substitue $\hat{\mathbf{c}}$ and $\hat{\mathbf{P}}$ obtained by the previous iteration to $\mathbf{c}$ and $\mathbf{P}$, respectively, and calculate

$$\begin{aligned}
\boldsymbol{\pi} &= \mathbf{w}'\mathbf{P} \\
\boldsymbol{\kappa} &= \mathbf{D}^{-1} + \boldsymbol{\pi}\mathbf{w} \\
\mathbf{k} &= \mathbf{P}\mathbf{w}\boldsymbol{\kappa}^{-1} \\
\hat{\mathbf{P}} &= \mathbf{P} - \mathbf{k}\boldsymbol{\pi} \\
\hat{\mathbf{c}} &= \mathbf{c} + \mathbf{k}\left(\mathbf{D}^{-1}\mathbf{d} - \mathbf{w}'\mathbf{c}\right)
\end{aligned}$$

Table 1: Summary of the proposed algorithm

$$\mathbf{D} = \mathbf{U}_{\hat{q}_t}^{-1} - \mathbf{U}_{q_t}^{-1} \tag{26}$$

$$\mathbf{d} = \mathbf{U}_{\hat{q}_t}^{-1}\boldsymbol{\mu}_{\hat{q}_t} - \mathbf{U}_{q_t}^{-1}\boldsymbol{\mu}_{q_t} \tag{27}$$

$$\mathbf{w} = [\underset{1}{0}, \ldots, 0, \underset{t}{1}, 0, \ldots, \underset{T}{0}]'. \tag{28}$$

$$\mathbf{D} = \Delta\mathbf{U}_{\hat{q}_t}^{-1} - \Delta\mathbf{U}_{q_t}^{-1} \tag{23}$$

$$\mathbf{d} = \Delta\mathbf{U}_{\hat{q}_t}^{-1}\Delta\boldsymbol{\mu}_{\hat{q}_t} - \Delta\mathbf{U}_{q_t}^{-1}\Delta\boldsymbol{\mu}_{q_t} \tag{24}$$

$$\mathbf{w} = [\underset{1}{0}, \ldots, 0, \underset{t-L}{w(-L)}, \ldots, \underset{t}{w(0)}, \ldots, \underset{t+L}{w(L)}, 0, \ldots, \underset{T}{0}]' \tag{25}$$

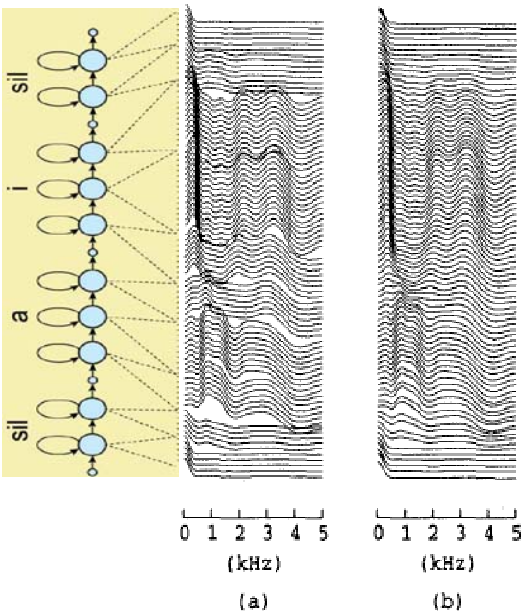


Figure 1: An example of parameter generation from an HMM composed by concatenation of phoneme models; sil, a, i, sil, (a) without dynamic feature, and (b) with dynamic feature. [1]

Fig.1.1 shows the spectra calculated from the 13 melcepstral coefficients generated by an HMM, which is composed by concatenation of the phoneme models for *sil, a, i, sil*.

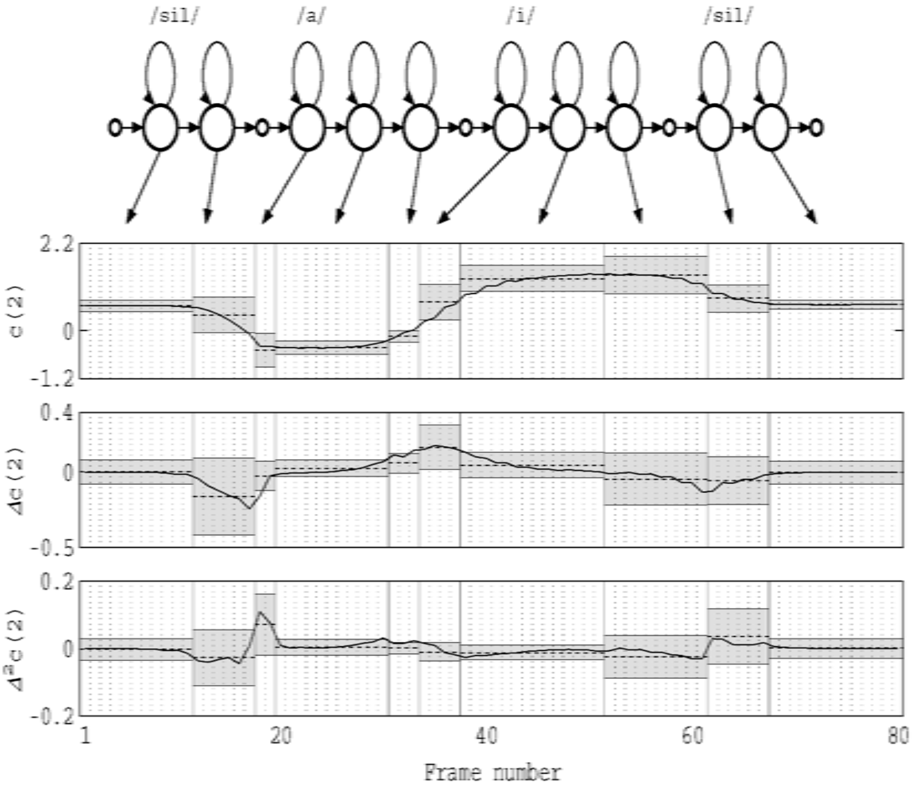


Fig.1.2 shows how, with the use of dynamic models, the curve of the concatenated phoneme models gets smoother. This results in a more natural speech.

2 Speech Parameter Generation Algorithms for HMM-Based Speech Synthesis [3]

2.1 Goal

The algorithm proposed in this paper is a little different to that on from Chapter 1. Previously the state sequence was assumed to be given or determined by a maximum likelihood criterion. However, in this paper we assume that at least part of the state sequence is hidden.

The goal of the algorithm is to determine the speech parameter vector sequence $O = \left[o_1^\top, o_2^\top, \dots, o_T^\top\right]^\top$ so that $P(O|\lambda) = \sum_{\text{all } Q} P(O, Q|\lambda$ is maximized with respect to **O**, where $Q = \{(q_1, i_1), (q_2, i_2), \dots, (q_T, i_T)\}$ is the state and mixture sequence, i.e. (q, i) indicates the i-th mixture of the state q.

Again $o_t = [c_t^\top, \Delta c_t^\top, \Delta^2 c_t^\top]^\top$ is the speech parameter vector. This time is consists of the static feature vector *ct* (e.g. cepstral coefficients) and the dynamic feature vectors *Δct* and *Δ²ct* (e.g. delta and delta-delta cepstral coefficients).

2.2 Derived Algorithm

The algorithm proposed in the paper is based on the EM algorithm, which is to fina critical point of the likelihood function.
An auxiliary function of the current and the new parameter vector sequence **O** and **O',** respectively is defined by $Q(O, O') = \sum_{\text{all } Q} P(O, Q|\lambda) \log P(O', Q|\lambda)$.

By substituting O' which maximizes the auxiliary function for O, the likelihood increases unless O is already a critical point of the likelihood. The equation can be rewritten as $Q(O, O') = P(O|\lambda)\left\{-\frac{1}{2}O'^\top \overline{U^{-1}} O' + O'^\top \overline{U^{-1}M} + \overline{K}\right\}$, where

$$\overline{U^{-1}} = \operatorname{diag}\left[\overline{U_1^{-1}}, \overline{U_2^{-1}}, \dots, \overline{U_T^{-1}}\right] \quad (19)$$

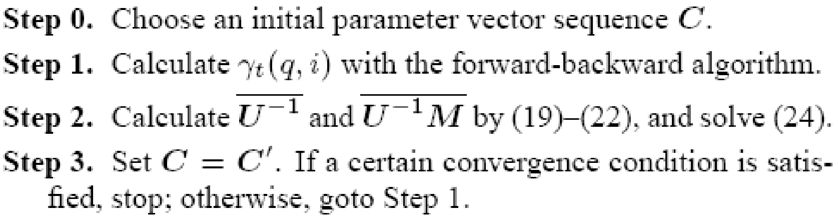
$$\overline{U_t^{-1}} = \sum_{q,i} \gamma_t(q,i) U_{q,i}^{-1} \quad (20)$$

$$\overline{U^{-1}M} = \left[\overline{U_1^{-1}\mu_1}^\top, \overline{U_2^{-1}\mu_2}^\top, \dots, \overline{U_T^{-1}\mu_T}^\top\right]^\top \quad (21)$$

$$\overline{U_t^{-1}\mu_t} = \sum_{q,i} \gamma_t(q,i)\, U_{q,i}^{-1} \mu_{q,i} \quad (22)$$

and $\gamma_t(q,i) = P(q_t = (q,i)|O,\lambda)$ is the occupancy probability, which canbe calculated by the forward-backward algorithm

Under the condition **O'=WC', C'** which maximizes Q(**O**,**O'**) is given by the following set of equations: $W^\top \overline{U^{-1}} W C' = W^\top \overline{U^{-1} M}$ This now can be solved using the recursive algorithm for dealing with the dynamic features from chapter 1.



Step 0. Choose an initial parameter vector sequence C.

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

Step 2. Calculate $\overline{U^{-1}}$ and $\overline{U^{-1}M}$ by (19)–(22), and solve (24).

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

Table 2: Summary of the algorithm used

2.3 Example

The example given for such an algorithm was the following. 5-state left-to-right HMMs were used, the state and mixture sequence is assumed to be hidden but the phoneme durations are given from phoneme duration densities. The speech was sampled at 16 kHz and windowed by a 25.6ms Blackman window with 5ms shift. The Mel-cepstral coefficients were obtained by a mel-cepstral analysis. The sentence fragment used was “kiNzokuhiroo.

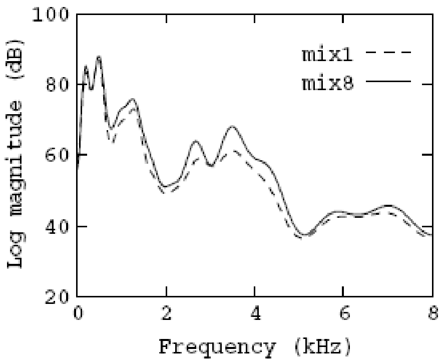


Figure 2: Spectra obtained from 1-mixture HMMs and 8-mixture HMMs.

Fig. 2.2 shows how with increasing mixtures the formant structures of the spoken sample get clearer. Informal listening tests have shown that the quality of the synthetic speech is considerably improved by increasing mixtures.

3 Multi-Space Probability Distribution HMM [4]

3.1 Problem

The question is why we need to use MSD – HMM. The problem is that we cannot apply both the conventional and continuous HMMs to observations which consist of continuous values and discrete symbols. For example in speech we often need to represent voiced and unvoiced signal

Thus we need a tool to use both of them. The solution is to use an extended HMM including discrete symbols (representing unvoiced signals) and continuous mixture HMMs (for voiced signals) as special cases.

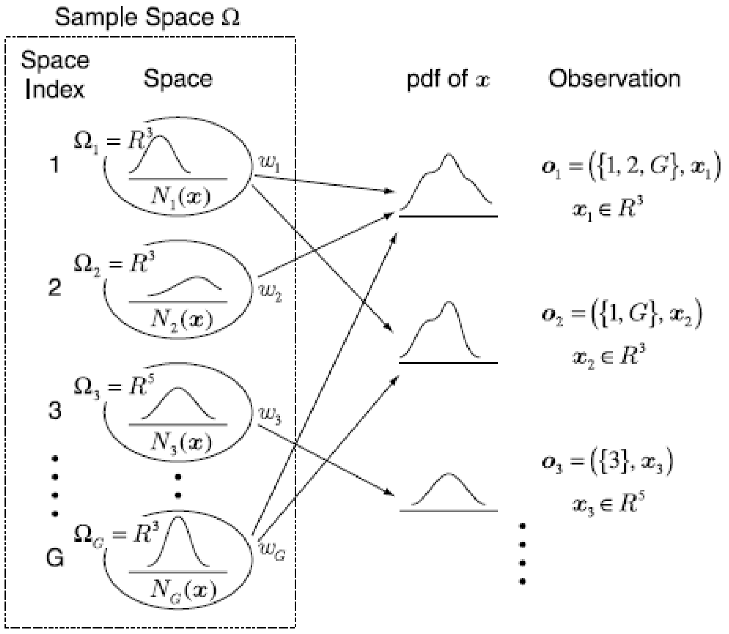
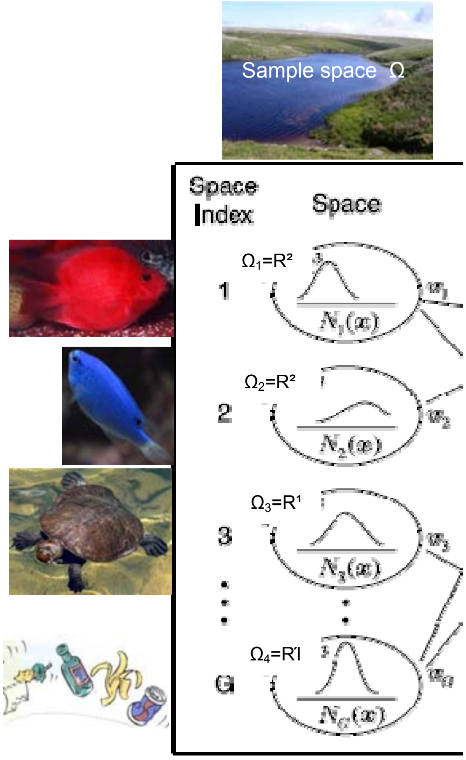


Fig. 1 Multi-space probability distribution and observation

The Figure above shows such a Multi-Space probability distribution. The Observation o1 consists of a set of space indices X1= {1, 2, G} and a 3D vector x1 ∈ R³. The vector x1 is drawn from one of three spaces Ω1, Ω2 or Ω3 of the sample space Ω. Thus its probability distribution function is w1N1(x) + w2N2(x) + wGNG(x).

3.2 Example

Let me give you a simple example for a MSD – HMM.
Imagine a man fishing in a pond.



In the pond which represents our sample space Ω there are red fish, blue fish, tortoises and rubbish.

Ω1 represents the 2D space for the length and height of the red fish, Ω2 stands for the 2D space for the length and height of the blue fish, Ω3 is the 1D space for the diameters of the tortoises given they are of circular shape and Ω4stands for the 0D space for articles of junk it thus represents discrete symbols.

The weights *w*1 to *w*4 are determined by the ratio of blue and red fish, tortoises and junk in the pond.

N1 and N2 are the 2D pdfs for the sizes and heights of the red and blue fish respectively, N3 is the 1D pdf for the diameter of the tortoises. N4, the pdf for the rubbish, is zero since the space has got no dimensionality.

Now, if the man catches a red fish he makes the following observation: **o** = ({1},**x**). This means he knows the fish is red, so the vector **x** is definitely of the vector space Ω1. If the same man would be fishing by night, he would not be able to see the colour of the fish. He could only determine the length or height of it. Thus the same catch by night would yield an observation **o** = ({1, 2},**x**).

3.3 Algorithm

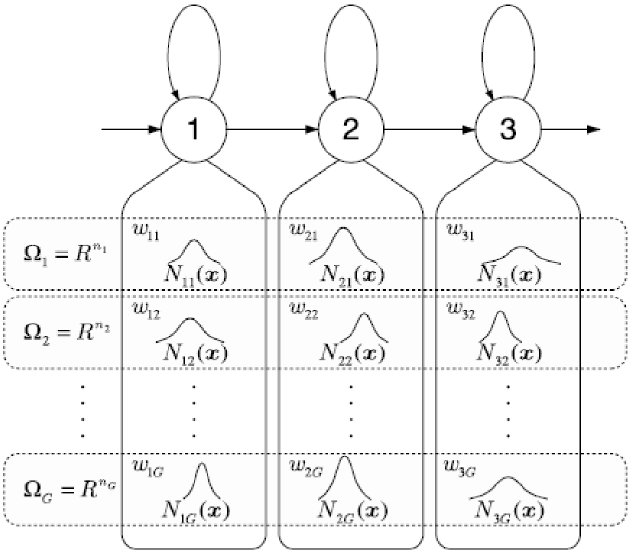


Fig. 2 An HMM based on multi-space probability distribution.

Each state i has got G pdfs Ni_G and their weights wi_G. The observation probability of **O,** P(**O**| λ) is calculated with the forward-backward algorithm
Then, we need to maximize the observation likelihood of P(**O**| λ). The reestimation formulas for the maximum likelihood estimation are calculated in analogy to the Baum-Welch-Algorithm.

4 Simultaneous Modelling of Spectrum, Pitch and Duration in HMM-based Speech Synthesis [5]

3.4 Simultaneous Modelling

In this approach a speech synthesis system is constructed in which spectrum pitch and state duration are modelled simultaneously in a unified framework HMM.
The feature vector consists of two streams, one for the spectral parameter vector and the other for the pitch parameter vector. Each phoneme HMM has its own state duration.

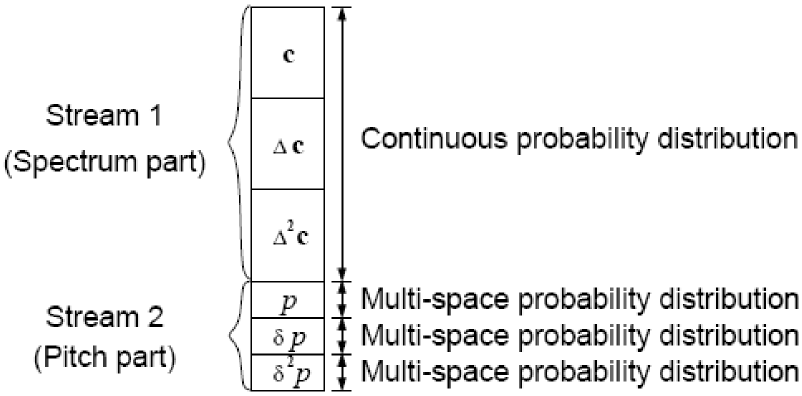

Stream 1
(Spectrum part)
c
Δc
Δ²c
Continuous probability distribution
Stream 2
(Pitch part)
p
δp
δ²p
Multi-space probability distribution
Multi-space probability distribution
Multi-space probability distribution

Figure 1. Feature vector.

The pitch patterns are modelled by HMMs based on multispace probability distribution since observation sequence composed of continuous values and discrete symbols and a normal HMM cannot handle both at the same time (see chapter 3).
The spectrum is modelled by continuous probability distributions.
State duration densities are modelled by single Gaussian distributions (dimension of the state duration density is equal to the number of the HMM states).

3.5 Context Dependent Model

Pitch, spectrum and duration is affected by many contextual factors. In this approach the following factors are taken into account:

- Mora[1] count of sentence
- Position of breath group in sentence
- Mora count of {preceding, current succeeding} breath group
- Position of current phoneme in current accentual phrase
- Mora count and accent type of {preceding, current succeeding} accentual phrase
- {preceding, current succeeding} part of speech
- Position of current phoneme in current accentual phrase
- {preceding, current succeeding} phoneme

The decision-tree based context clustering is applied because contextual factors increase and thus their combinations increase exponentially Therefore model parameters cannot be estimated with sufficient accuracy by limited training data. It is impossible to prepare speech database which includes all combinations of contextual factors.

Since spectrum, pitch and duration have their own influential contextual factors the distributions of the respective parameters are clustered independently.

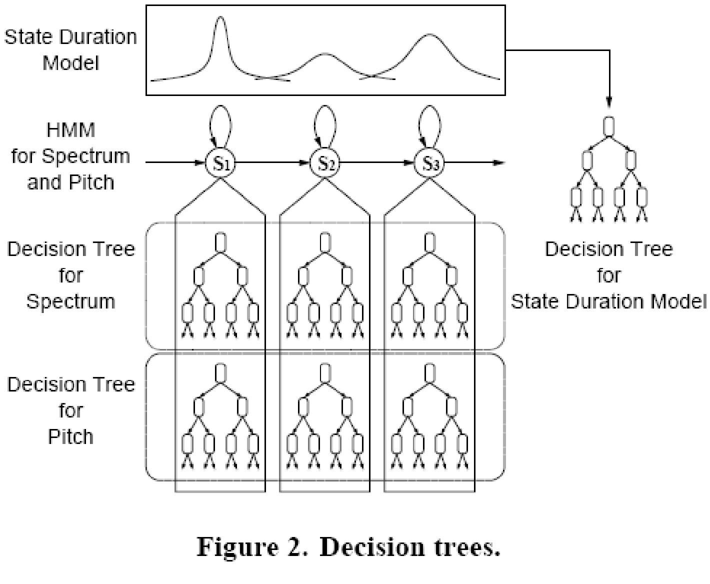

State Duration Model
HMM for Spectrum and Pitch
S1
S2
S3
Decision Tree for Spectrum
Decision Tree for Pitch
Decision Tree for State Duration Model

Figure 2. Decision trees.

3.6 Text-to-Speech Synthesis System

This is how the synthesis part works:

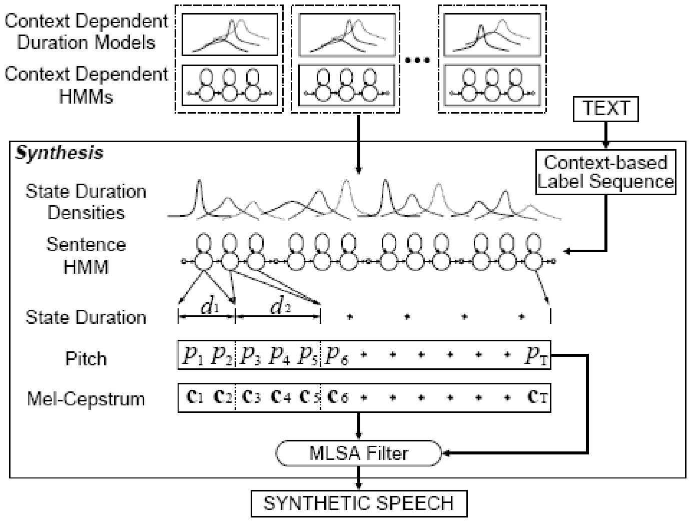

Context Dependent Duration Models
Context Dependent HMMs
TEXT
Synthesis
Context-based Label Sequence
State Duration Densities
Sentence HMM
State Duration
d_1
d_2
Pitch
p_1 p_2 p_3 p_4 p_5 p_6 p_T
Mel-Cepstrum
c_1 c_2 c_3 c_4 c_5 c_6 c_T
MLSA Filter
SYNTHETIC SPEECH

Figure 3. Synthesis part of the system.

An arbitrary text is converted to a context-based label sequence by concatenated context dependent HMMs, the state durations of the sentence HMM is the constructed according to label sequence.

The State durations are determined so as to maximize the likelihood of the state duration densities.

The Sequence of mel-cepstrum coefficients and pitch values are generated.

Finally, the speech is synthesized directly from the mel-cepstrum coefficients and pitch values by an MLSA filter.

5 References

[1] K. Tokuda, T. Kobayashi and S. Imai, Speech Parameter Generation from HMM using Dynamic Features, Proc. ICASSP, 1995

[2] http://hts.sp.nitech.ac.jp/?Publications – see ‘Attach file’ → tokuda_TTSworkshop2002.pdf

[3] Tokuda, Yoshimura, Masuko, Kobayashi, Kitamura, Speech Parameter Generation Algortihms for HMM-based Speech Synthesis, ICASSP, 2000

[4] Tokuda, Masuko, Miyazaki, Kobayashi, Multi-Space Probability Distribution HMM, IEICE Trans. Inf. & Syst., 2000

[5] Yoshimura, Tokuda, Masuko, Kobayashi, Kitamura, Simultaneous modelling of Spectrum, Pitch and Duration in HMM-based Speech Synthesis, Proc EU-ROSPEECH, 1999