ON THE STATE DEFINITION FOR A TRAINABLE EXCITATION MODEL IN HMM-BASED SPEECH SYNTHESIS.

R. Maia[†,‡], *T. Toda*[†,††], *K. Tokuda*[†,††‡], *S. Sakai*[†,‡], *S. Nakamura*[†,‡]

[†]National Inst. of Inform. and Comm. Technology (NICT), Japan
[‡]ATR Spoken Language Comm. Labs, Japan
[††]Nara Institute of Science and Technology, Japan
[††‡]Nagoya Institute of Technology, Japan
{ranniery.maia,shinsuke.sakai,satoshi.nakamura}@atr.jp
tomoki@is.naist.jp,tokuda@nitech.ac.jp

ABSTRACT

One of the issues of speech synthesizers based on hidden Markov models concerns the *vocoded quality* of the synthesized speech. From the principle of analysis-by-synthesis speech coders a trainable excitation model has been proposed to improve naturalness, where the method consists in the design of a set of state-dependent filters in a way to minimize the distortion between residual and synthetic excitation. Although this approach seems successful, state definition still represents an open issue. This paper describes a method for state definition wherein bottom-up clustering is performed on full context decision trees, using the likelihood of the residual database as merging criterion. Experiments have shown that improvement on residual modeling through better filter design can be achieved.

Index Terms— Speech processing, speech synthesis, hidden Markov models, digital filters.

1. INTRODUCTION

In the past years some attempts have been made to improve the naturalness of hidden Markov model (HMM)-based speech synthesizers. This subject has gained attention from the speech synthesis research community due to the fact that once synthesizers based on this technology achieve quality similar to unit concatenation-based systems, they might finally suit the increasing demand for high-quality speech synthesis with flexibility concerning the possibility of voice transformation, utilization of small corpora and footprint, etc.

Many approaches have been proposed to improve the quality of HMM-based speech synthesizers through the design of better excitation models, e.g [1, 2, 3, 4, 5]. Most of them are based on the modeling of auxiliary parameters in the HMMs themselves so that during the synthesis a parametric excitation signal can be produced. In Yoshimura's approach [1] for instance, parameters encoded by the Mixed Excitation Linear Prediction (MELP) algorithm [6] are used to construct an excitation signal in the same way as performed by the MELP decoder. Using the same philosophy, Zen et al proposed the utilization of the STRAIGHT vocoding method for HMM-based speech synthesis [2]. Going beyond the source-filter framework, modeling of sinusoidal coefficients was utilized by Abdel-Hamid [4] and harmonic plus noise model by Hemptinne [5]. Back to the source-filter scheme, Cabral proposed a glottal source model to replace the input pulse train during the synthesis [7]. In Cabral's method glottal source parameters are derived from the speech database itself, with no modeling by HMMs.

In [8] a trainable excitation model for HMM-based speech synthesis is described. The method is based on the principle of analysis-by-synthesis speech coders and consists in the optimization of some state-dependent filter coefficients through the minimization of the difference between synthetic excitation and residual, with the latter being directly obtained from the speech corpus through inverse filtering. Although the scheme in question performs well, state definition remains vague. Specifically for the experiments presented in [8], filter states were regarded as leaves of decision trees for mel-cepstral coefficients, constructed with the utilization of phonetic questions. Eventually, the resulting clusters were used to tag Viterbi-aligned segments of the training database. Therefore, as reported in [8], since this method is rather empirical, an appropriate state definition still represents an open issue. This paper presents an approach to address this problem. The proposed algorithm performs bottom-up clustering in the usual full context mel-cepstral coefficients decision trees, generated during the training of the HMM-based synthesizer, using the likelihood of residual sequences as merging criterion.

The rest of this paper is organized as follows: Section 2 outlines the trainable excitation model of [8]; Section 3 concerns state definition, starting with the description of how it has been performed followed by the description of the proposed method; Section 4 shows some experiments; and the conclusions are in Section 5.

2. TRAINABLE EXCITATION

2.1. Synthesis part

Fig. 1 depicts the synthesis stage of the excitation model proposed in [8], where pulse train, $t(n)$, and white noise, $w(n)$, are filtered through $H_v(z)$ and $H_u(z)$, respectively, and added together to result in the excitation signal $\tilde{e}(n)$. The voiced and unvoiced filters, $H_v(z)$ and $H_u(z)$, respectively, are associated with each HMM state position s and their transfer functions are

$$H_v^s(z) = \sum_{l=-M/2}^{M/2} h_s(l) z^{-l}, \tag{1}$$

$$H_u^s(z) = \frac{K_s}{1 - \sum_{l=1}^{L} g_s(l) z^{-l}}, \tag{2}$$

where M and L are the respective orders.

The design of the voiced filter $H_v(z)$ is performed in a way that the voiced excitation $v(n)$ becomes as close as possible to residual

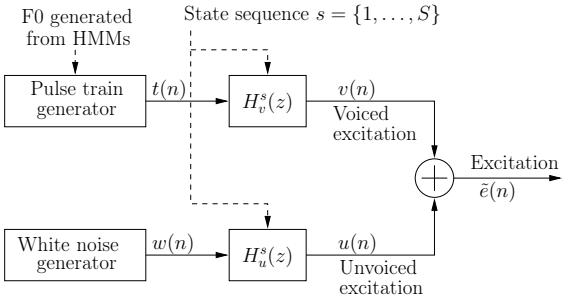


Fig. 1. During the synthesis: filters $H_v(z)$ and $H_u(z)$ are associated with each HMM state s.

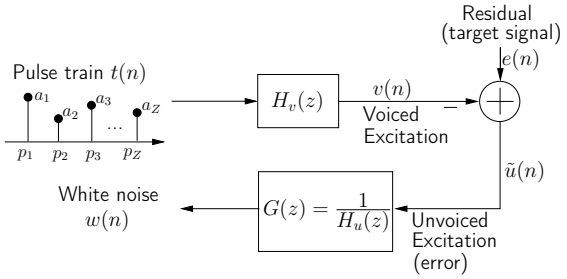


Fig. 2. During the training: pulse train and residual are the input while white noise is the assumed output.

sequences in voiced regions. The unvoiced filter $H_u(z)$, on the other hand, weights the input noise sequence $w(n)$ in order to produce the unvoiced component, $u(n)$, of the excitation signal $\tilde{e}(n)$.

2.2. Training part

The excitation model components, namely the voiced and unvoiced filters $H_v(z)$ and $H_u(z)$, and pulse train $t(n)$, are iteratively calculated in a way to minimize the error between residual and synthetic excitation. In order to visualize the procedure, the block diagram of Fig. 2 should be taken into account. This block can be obtained from the one shown in Fig. 1 if we consider the residual, $e(n)$, as the input and white noise, $w(n)$, as the output. By making an analogy with analysis-by-synthesis speech coders [6], one can notice that the target signal is represented by $e(n)$, the error of the system is $Kw(n)$, and the terms whose incremental modification can minimize the power of $Kw(n)$[1] are the filters and pulse train. Therefore, the problem of achieving an excitation signal whose waveform can be as close as possible to the residual consists of the design of $H_v(z)$ and $H_u(z)$, and optimization of $t(n)$.

2.2.1. Filter determination

Using matrices and vectors, with N being the total number of samples of the entire database, the filters are determined by minimizing the mean squared error ε, given by

$$\varepsilon = \frac{1}{N}\left[\mathbf{e} - \sum_{s=1}^{S}\mathbf{A}_s\mathbf{h}_s\right]^T \mathbf{G}^T\mathbf{G}\left[\mathbf{e} - \sum_{s=1}^{S}\mathbf{A}_s\mathbf{h}_s\right], \qquad (3)$$

where $\mathbf{G}$ is an $N \times N$ matrix containing the impulse response of the inverse unvoiced filter $G(z)$, $\mathbf{h}_s = [h_s(-M/2)\cdots h_s(M/2)]^T$ is the impulse response vector of the voiced filter for state s, and the term $\mathbf{A}_s$ is the overall pulse train matrix where only pulse positions belonging to state s are non-zero. In this case, each state $s = \{1, \ldots, S\}$ corresponds to a different HMM state position covering the entire database, after Viterbi-alignment.

Voiced filter coefficients for a given state s can be obtained by making $\partial\varepsilon/\partial\mathbf{h}_s = 0$, which results in a linear system for the solution of $\mathbf{h}_s$ [8]. On the other hand, the unvoiced filter coefficients for state s, $\{g_s(1), \ldots, g_s(L)\}$, and related gain K_s, are determined by performing linear prediction analysis on the unvoiced excitation signal $\tilde{u}(n) = e(n) - v(n)$ over segments tagged as state s.

2.2.2. Pulse optimization

Aside from the determination of the filters, the positions and amplitudes of $t(n)$, $\{p_1, \ldots, p_Z\}$ and $\{a_1, \ldots, a_Z\}$, with Z being the number of pulses of the entire training database, are modified in the sense of minimizing the mean squared error of (3). The process in which the positions and amplitudes are calculated resembles multipulse excitation linear prediction coding algorithms [6].

2.2.3. Recursive algorithm

The overall procedure for the design of the filters and optimization of $t(n)$ is performed in an interchanging way, with the convergence criterion being either filter coefficient variation or mean squared error reduction.

3. STATE DEFINITION

3.1. Phonetic decision trees

In the experiments presented in [8] states $s = \{1, \ldots, S\}$ were regarded as leaves of decision trees specifically constructed to tag the HMM states eventually used to train the excitation model. The trees were constructed using solely phonetic questions, and the factor used as stopping criterion for the splitting process was set in a way that just gross phonetic information, such as voiced, unvoiced, fricative, stops, etc, was conveyed by the trees.

3.2. New approach for state-definition: bottom-up clustering

3.2.1. The idea

Since the idea of utilizing phonetic decision trees seems rather empirical, it is not guaranteed whether it could work effectively across different database sizes and different languages, with possibly very few or large number of phonetic questions for clustering the features usually employed in the HMM-based speech synthesis technique.

To obtain a less empirical state definition, processing of the trees for mel-cepstral coefficients which is actually used by the HMM-based synthesizer could be a good choice. In fact, the initial idea of states for the excitation model corresponded to leaves of the trees in question because of the direct relationship between mel-cepstral coefficients and residual. However, as the number of clusters might be considerably large depending on several factors, the number of residual segments with similar characteristics for some terminal nodes might not be enough in order to enable a robust design of the voiced filter impulse responses. In order words, the size of the trees which might be adequate for modeling the distribution of mel-cepstral coefficients, may not be for the calculation of the filters. Therefore, merging the leaves of the referred trees according to some criterion related to the residual database, in connection with the filters themselves, may lead to an effective approach. The maximization of the

[1] $w(n)$ itself is assumed to have power one.

likelihood of residual signals $e(n)$ given the excitation model is perhaps a good criterion for the bottom-up clustering in question.

The procedure above pictured presents two main advantages when comparing with the utilization of phonetic decision trees: (1) the filter states correspond to a more general version of the trees which are actually employed for HMM-based speech synthesis; (2) the algorithm can automatically define the states, assuming likelihood increment or number of final clusters as stopping criterion.

3.2.2. Merging criterion: residual likelihood

Assuming that the noise sequence $w(n)$ which drives the unvoiced filter $H_u(z)$ is Gaussian, the log likelihood of the output vector $\mathbf{u}$ is

$$\log P[\mathbf{u}|\mathbf{H}_u] = -\frac{N}{2}\log 2\pi + \frac{1}{2}\log|\mathbf{G}^T\mathbf{G}| - \frac{1}{2}\mathbf{u}^T\mathbf{G}^T\mathbf{G}\mathbf{u}. \quad (4)$$

Since

$$|\mathbf{G}^T\mathbf{G}|^{-1} = \prod_{n=0}^{N-1} \frac{K^2}{\left|1 - \sum_{l=1}^{L} g(l)e^{-j\omega_n l}\right|^2}, \quad (5)$$

the second term of (4) becomes

$$\frac{1}{2}\log|\mathbf{G}^T\mathbf{G}| = \frac{1}{2}\sum_{n=0}^{N-1}\log\left|1 - \sum_{l=1}^{L} g(l)e^{j w_n l}\right|^2 - N\log K. \quad (6)$$

Because $G(z)$ is minimum-phase, the first term of (6) is zero [9] (pages 129-130). Further, it can be noticed for the third element of (4) that

$$\mathbf{u}^T\mathbf{G}^T\mathbf{G}\mathbf{u} = K^2\mathbf{w}^T\mathbf{w} = K^2\sum_{n=0}^{N-1} w^2(n). \quad (7)$$

Assuming that $w(n)$ is white with mean zero and variance one, then

$$E\{w^2(n)\} = 1 \Rightarrow \frac{1}{N}\sum_{n=0}^{N-1} w^2(n) \approx 1, \quad (8)$$

and consequently

$$\mathbf{u}^T\mathbf{G}^T\mathbf{G}\mathbf{u} = NK^2. \quad (9)$$

Thus, the likelihood of $e(n)$ given the excitation model[2] depends only on the unvoiced filter gain K,

$$\log P[\mathbf{e}|\mathbf{H}_v, \mathbf{H}_u, \mathbf{t}] = -\frac{N}{2}\log 2\pi - N\left(\log K + \frac{K^2}{2}\right). \quad (10)$$

3.2.3. State-dependency

Since filter coefficients are different for each state, (10) can be rewritten as

$$\log P[\mathbf{e}|\mathbf{H}_v, \mathbf{H}_u, \mathbf{t}] = -\frac{N}{2}\log 2\pi + \sum_{s=1}^{S}\mathcal{L}_s, \quad (11)$$

where

$$\mathcal{L}_s = -N_s\left(\log K_s + \frac{K_s^2}{2}\right), \quad (12)$$

is the likelihood contribution yielded by state s, N_s is the respective total number of samples, and K_s is the corresponding unvoiced filter gain. From (12) one can see that the smaller the gain factor K_s is, the greater is the contribution of the state to the overall likelihood. Further, one can also notice that small K_s means that the power of the unvoiced excitation $\tilde{u}(n) = e(n) - v(n)$ of segments belonging to state s is small, which finally bring us to conclude that the voiced filter $H_v(z)$ is performing well in terms of modeling the residual $e(n)$ through the voiced excitation $v(n)$.

[2]Note that $P[\mathbf{u}|\mathbf{H}_u] \Leftrightarrow P[\mathbf{e}|\mathbf{H}_v, \mathbf{H}_u, \mathbf{t}]$.

3.2.4. Clustering algorithm

The procedure bellow is utilized for the clustering process. It starts assuming the existence of S' clusters from initial decision trees. The desired number of final clusters is S. For each merging step:

1. calculate all the possible $\mathcal{L}_{\text{inc}}$, where
$$\mathcal{L}_{\text{inc}} = \mathcal{L}_{s_i,s_j} - \mathcal{L}_{s_i} - \mathcal{L}_{s_j}, \quad (13)$$
with $\mathcal{L}_{s_i,s_j}$ being the likelihood of the cluster resulted by the merging of s_i and s_j;
2. merge clusters s_i and s_j with biggest $\mathcal{L}_{\text{inc}}$;
3. make $S' = S' - 1$;
4. if $S' = S$ (or if $\mathcal{L}_{\text{inc}}$ falls below a given threshold), stop. Otherwise, go to the next merging step.

4. EXPERIMENTS

In order to verify the effectiveness of the bottom-up clustering method, the ATR503 Japanese speech database was used to train an HMM-based synthesizer and two excitation models.

4.1. Excitation models

4.1.1. Using states defined by phonetic decision trees

The first excitation model was trained assuming the leaves of phonetic decision trees for mel-cepstral coefficients as states, constructed according to the description of Section 3.1, i.e., using only phonetic questions and large MDL (Minimum Description Length) factor, $\lambda = 10$. A total of $S = 75$ clusters were created.

4.1.2. Using states defined by bottom-up clustering

The second excitation model was derived according to the bottom-up clustering approach. The initial trees were the ones constructed for the distribution of mel-cepstral coefficients in HMM-based speech synthesis. The stopping criterion for the merging process was chosen as $S = 75$, in order to compare with the states defined by phonetic decision trees of Section 4.1.1.

4.2. Result of the clustering process

Table 1 summarizes the result of the clustering procedure whereas Fig. 3 shows the evolution of the likelihood increment $\mathcal{L}_{\text{inc}}$ for each merging step. It can been seen that for the first 114 merging steps the likelihood increment is positive, even though it would be expected that by merging two clusters it should be negative due to the reduction of degrees of freedom for modeling the same amount of data. However, by merging two clusters whose voiced filters were not robustly calculated may produce a better resulting filter if the overall number of similar segments increase against the number of different segments within the merged cluster. With better filters, the power of the error signal $\tilde{u}(n) = e(n) - h(n) * t(n)$ is decreased, increasing the likelihood. From merging step 115, when this problem is apparently solved, $\mathcal{L}_{\text{inc}}$ becomes negative.

Table 1. Summary of the clustering process.

Number of initial clusters	699
Number of final clusters	75
Initial clusters which were not merged	9
Clusters created through the merging of initial ones	66

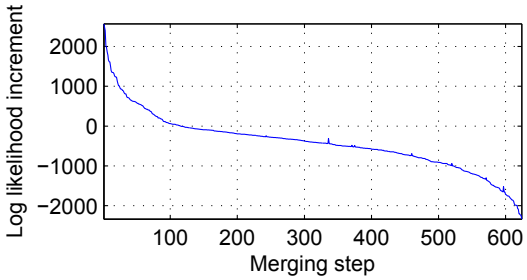


Fig. 3. Evolution of the likelihood increment $\mathcal{L}_{\text{inc}}$ across the merging steps.

4.3. Filter impulse responses

A problem of utilizing phonetic decision trees for state definition is that depending on the database some voiced filters can show poor modeling properties, as it can be seen in the 3-D depiction of the voiced filter impulse responses of Fig. 4(a). This happens because sometimes residual segments with very different characteristics, such as voiced fricatives and unvoiced stops, may belong to the same cluster, and consequently their modeling through the convolution of voiced filters and pulse trains becomes difficult. This problem is significantly alleviated with the utilization of the states defined by the bottom-up clustering approach, as shown in Fig. 4(b).

4.4. Likelihood of the excitation models

Table 2 shows the likelihood of $e(n)$ given the excitation models, calculated according to (11). The higher likelihood for the bottom-up state definition approach means that voiced excitation signals $v(n)$ are closer to the target signals $e(n)$ in the analysis-by-synthesis system of Fig. 2. Consequently, it can be concluded that better modeling of the residual database is achieved by the method in question.

5. CONCLUSION

This paper presented a new direction for state definition in a trainable excitation model for HMM-based speech synthesis. The method performs bottom-up clustering on full context decision trees for mel-cepstral coefficients using the residual likelihood maximization criterion. Experiments have shown that states defined according to this approach result in better residual modeling.

6. REFERENCES

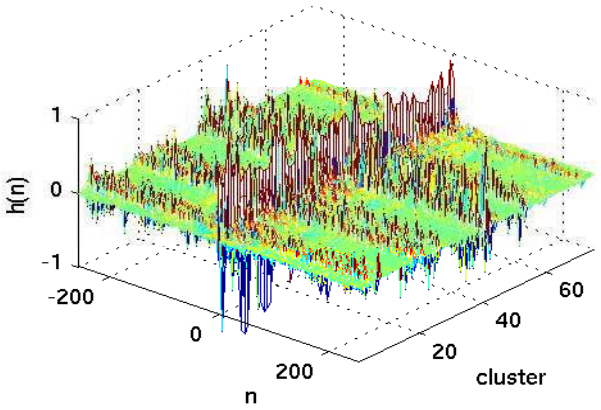
[1] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Mixed-excitation for HMM-based speech synthesis," in *Proc. of EUROSPEECH*, 2001.

[2] H. Zen, T. Toda, M. Nakamura, and K. Tokuda, "Details of the Nitech HMM-based speech synthesis for Blizzard Challenge 2005," *IEICE Trans. on Inf. and Systems*, vol. E90-D, Jan. 2007.

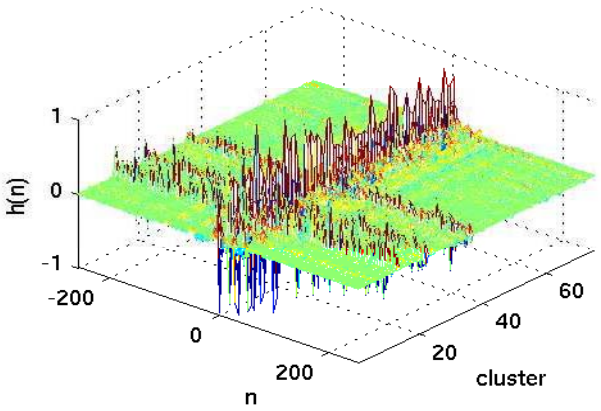
[3] S. J. Kim and M. Hahn, "Two-band excitation for HMM-based speech synthesis," *IEICE Trans. Inf. & Syst.*, vol. E90-D, Jan. 2007.

Table 2. Likelihood of the residual database for two different state configurations for the excitation model.

State definition	$\mathcal{L}$
By phonetic decision trees	-15,154,631.27
By bottom-up clustering	-14,979,007.86



(a) Filters derived according to states of phonetic decision trees.



(b) Filters derived according to states defined by bottom-up clustering.

Fig. 4. Impulse responses of $H_v^s(z)$ for $s = \{1, \ldots, 75\}$.

[4] O. Abdel-Hamid, S. Abdou, and M. Rashwan, "Improving the Arabic HMM based speech synthesis quality," in *Proc. of ICSLP*, 2006.

[5] C. Hemptinne, "Integration of the Harmonic plus Noise Model (HNM) into the Hidden Markov Model-based speech synthesis system (HTS)," M.S. thesis, IDIAP, June 2006.

[6] W. Chu, *Speech Coding Algorithms*, Wiley-Interscience, 2003.

[7] J. Cabral, S. Renals, K. Richmond, and J. Yamagishi, "Towards an improved modeling of the glottal source in statistical parametric speech synthesis," in *Proc. of SSW6*, 2007.

[8] R. Maia, T. Toda, H. Zen, Y. Nankaku, and K. Tokuda, "A trainable excitation model for HMM-based speech synthesis," in *Proc. of INTERSPEECH*, 2007.

[9] J. D. Markel and A. H. Gray, Jr., *Linear prediction of speech*, Springer-Verlag, 1986.