Available online at www.sciencedirect.com
ScienceDirect
Speech Communication 51 (2009) 1039–1064
SPEECH COMMUNICATION
www.elsevier.com/locate/specom

Review

Statistical parametric speech synthesis

Heiga Zen [a,b,*], Keiichi Tokuda [a], Alan W. Black [c]

[a] Department of Computer Science and Engineering, Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya 466-8555, Japan
[b] Cambridge Research Laboratory, Toshiba Research Europe Ltd., 208 Cambridge Science Park, Milton Road, Cambridge CB4 0GZ, UK
[c] Language Technologies Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA

Received 14 January 2009; received in revised form 6 April 2009; accepted 8 April 2009

Abstract

This review gives a general overview of techniques used in *statistical parametric speech synthesis*. One instance of these techniques, called hidden Markov model (HMM)-based speech synthesis, has recently been demonstrated to be very effective in synthesizing acceptable speech. This review also contrasts these techniques with the more conventional technique of unit-selection synthesis that has dominated speech synthesis over the last decade. The advantages and drawbacks of statistical parametric synthesis are highlighted and we identify where we expect key developments to appear in the immediate future.

Keywords: Speech synthesis; Unit selection; Hidden Markov models

Contents

* Corresponding author. Address: Cambridge Research Laboratory, Toshiba Research Europe Ltd., 208 Cambridge Science Park, Milton Road, Cambridge CB4 0GZ, UK. Tel.: +44 1223 436 975; fax: +44 1223 436 909.
E-mail addresses: heiga.zen@crl.toshiba.co.uk (H. Zen), tokuda@nitech.ac.jp (K. Tokuda), awb@cs.cmu.edu (A.W. Black).

0167-6393/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.specom.2009.04.004

1. Background

With the increase in the power and resources of computer technology, building natural-sounding synthetic voices has progressed from a knowledge-based approach to a data-based one. Rather than manually crafting each phonetic unit and its applicable contexts, high-quality synthetic voices may be built from sufficiently diverse single-speaker databases of natural speech. We can see progress from fixed inventories, found in diphone systems (Moulines and Charpentier, 1990) to more general, but more resource consuming, techniques of unit-selection synthesis where appropriate sub-word units are automatically selected from large databases of natural speech (Hunt and Black, 1996).

ATR ν-talk was the first to demonstrate the effectiveness of the automatic selection of appropriate units (Sagisaka et al., 1992), then CHATR generalized these techniques to multiple languages and an automatic training scheme (Hunt and Black, 1996). Unit-selection techniques have evolved to become the dominant approach to speech synthesis. The quality of output derives directly from the quality of recordings, and it appears that the larger the database the better the coverage. Commercial systems have exploited these techniques to bring about a new level of synthetic speech (Breen and Jackson, 1998; Donovan and Eide, 1998; Beutnagel et al., 1999; Coorman et al., 2000).

However, although certainly successful, there is always the issue of spurious errors. When a required sentence happens to need phonetic and prosodic contexts that are under-represented in a database, the quality of the synthesizer can be severely degraded. Even though this may be a rare event, a single bad join in an utterance can ruin the listeners' flow. It is not possible to guarantee that bad joins and/or inappropriate units will not occur, simply because of the vast number of possible combinations that could occur. However, it is often possible to almost always avoid these for particular applications. Limited domain synthesizers (Black and Lenzo, 2000), where the database has been designed for the particular application, go a long way toward optimizing almost all synthetic output. Despite the objective for optimal synthesis all the time, there are limitations in unit-selection techniques. As no (or few) modifications to the selected pieces of natural speech are usually done, this limits the output speech to the same style as that in the original recordings. With the need for more control over speech variations, larger databases containing examples of different styles are required. IBM's stylistic synthesis (Eide et al., 2004) is a good example but this is limited by the number of variations that can be recorded. Unfortunately, recording large databases with variations is very difficult and costly (Black, 2003).

In direct contrast to this selection of actual instances of speech from a database, statistical parametric speech synthesis has also grown in popularity over the last years (Yoshimura et al., 1999; Ling et al., 2006; Black, 2006; Zen et al., 2007c). Statistical parametric synthesis might be most simply described as generating the *average* of some sets of similarly sounding speech segments. This contrasts directly with the target in unit-selection synthesis that retains natural unmodified speech units, but using parametric models offers other benefits. In both the Blizzard Challenge in 2005 and 2006 (Tokuda and Black, 2005; Bennett, 2005; Bennett and Black, 2006), where common speech databases were provided to participants to build synthetic voices, the results from subjective listening tests revealed that one instance of statistical parametric synthesis techniques offered synthesis that was more preferred (through mean opinion scores) and more understandable (through word error rates) (Ling et al., 2006; Zen et al., 2007c). Although even the proponents of statistical parametric synthesis feel that the best examples of unit-selection synthesis are better than the best examples of statistical parametric synthesis, overall it appears that the quality of statistical parametric synthesis has already reached a level where it can stand in its own right. The quality issue comes down to the fact that, given a parametric representation, it

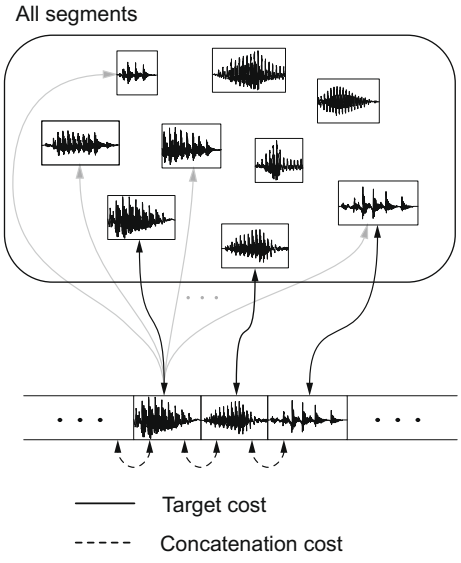

Fig. 1. Overview of general unit-selection scheme. Solid lines represent target costs and dashed lines represent concatenation costs.

is necessary to reconstruct the speech from these parameters. The process of reconstruction is still not ideal. Although modeling the spectral and prosodic features is relatively well defined, models of residual/excitation have yet to be fully developed, even though composite models like STRAIGHT (Kawahara et al., 1999) are proving to be useful (Irino et al., 2002; Zen et al., 2007c).

The aim of this review is to give a general overview of techniques in statistical parametric speech synthesis. Although many research groups have contributed to progress in statistical parametric speech synthesis, the description given here is somewhat biased toward implementation of the HMM-based speech synthesis system (HTS)[1] (Yoshimura et al., 1999; Zen et al., 2007b) for the sake of logical coherence.

The rest of this review is organized as follows. First, a more formal definition of unit-selection synthesis that allows easier contrast with statistical parametric synthesis is described. Then, the core architecture of statistical parametric speech synthesis is more formally defined, specifically based on the implementation of HTS. The following sections discuss some of the advantages and drawbacks in a statistical parametric framework, highlighting some possible directions to take in the future. Various refinements that are needed to achieve state-of-the-art performance are also discussed. The final section discusses conclusions we drew along with some general observations and a discussion.

2. Unit-selection synthesis

The basic unit-selection premise is that we can synthesize new natural-sounding utterances by selecting appropriate sub-word units from a database of natural speech.

There seem to be two basic techniques in unit-selection synthesis, even though they are theoretically not very different. Hunt and Black presented a selection model (Hunt and Black, 1996), described in Fig. 1, which actually existed previously in ATR ν-talk (Sagisaka et al., 1992). The basic notion is that of a *target cost*, i.e., how well a candidate unit from the database matches the required unit, and a *concatenation cost*, which defines how well two selected units combine. The definition of target cost between a candidate unit, u_i, and a required unit, t_i, is

$$C^{(t)}(t_i, u_i) = \sum_{j=1}^{p} w_j^{(t)} C_j^{(t)}(t_i, u_i), \tag{1}$$

where j indexes over all features (phonetic and prosodic contexts are typically used). The concatenation cost is defined as

$$C^{(c)}(u_{i-1}, u_i) = \sum_{k=1}^{q} w_k^{(c)} C_k^{(c)}(u_{i-1}, u_i), \tag{2}$$

[1] Available for free download at the HTS website (Tokuda et al., 2008). This includes recipes for building state-of-the-art speaker-dependent and speaker-adaptive synthesizers using CMU ARCTIC databases (Kominek and Black, 2003), which illustrate a number of the approaches described in this review.

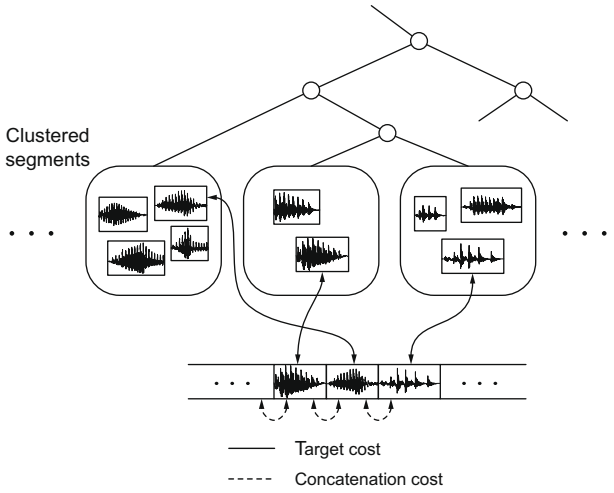

Fig. 2. Overview of clustering-based unit-selection scheme. Solid lines represent target costs and dashed lines represent concatenation costs.

where k, in this case, may include spectral and acoustic features. These two costs must then be optimized to find the string of units, $u_{1:n} = \{u_1, \ldots, u_n\}$, from the database that minimizes the overall cost, $C(t_{1:n}, u_{1:n})$, as

$$\hat{u}_{1:n} = \arg\min_{u_{1:n}} \{C(t_{1:n}, u_{1:n})\}, \tag{3}$$

where

$$C(t_{1:n}, u_{1:n}) = \sum_{i=1}^{n} C^{(t)}(t_i, u_i) + \sum_{i=2}^{n} C^{(c)}(u_{i-1}, u_i). \tag{4}$$

The second direction, described in Fig. 2, uses a clustering method that allows the target cost to effectively be pre-calculated (Donovan and Woodland, 1995; Black and Taylor, 1997). Units of the same type are clustered into a decision tree that asks questions about features available at the time of synthesis (e.g., phonetic and prosodic contexts).

There has been, and will continue to be, a substantial amount of work on looking at what features should be used, and how to weigh them. Getting the algorithms, measures, and weights right will be the key to obtaining consistently high-quality synthesis. These cost functions are formed from a variety of heuristic or ad hoc quality measures based on the features of the acoustic signal and given texts. Target-cost and concatenation-cost functions based on statistical models have recently been proposed (Mizutani et al., 2002; Allauzen et al., 2004; Sakai and Shu, 2005; Ling and Wang, 2006). Weights ($w_j^{(t)}$ and $w_k^{(c)}$) have to be found for each feature, and actual implementations use a combination of trained and manually tuned weights. All these techniques depend on an *acoustic distance measure* that is taken to be correlated with human perception.

Work on unit-selection synthesis has investigated the optimal size of units to be selected. The longer the unit, the larger the database must generally be to cover the required domain. Experiments with different-sized units tend to demonstrate that small units can be better as they offer more potential join points (Kishore and Black,

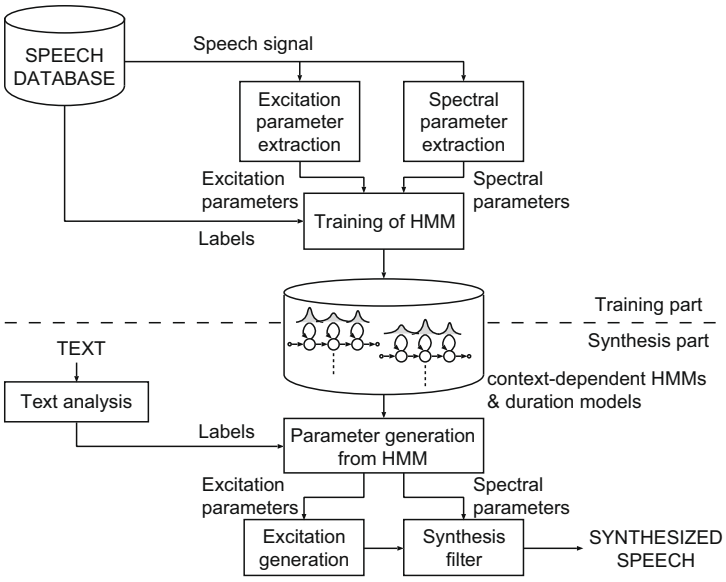

Fig. 3. Block-diagram of HMM-based speech synthesis system (HTS).

2003). However, continuity may also be affected with more join points. Various publications have discussed the superiority of different-sized units, i.e., from frame-sized (Hirai and Tenpaku, 2004; Ling and Wang, 2006), HMM state-sized (Donovan and Woodland, 1995; Huang et al., 1996), half-phones (Beutnagel et al., 1999), diphones (Black and Taylor, 1997), to much larger and even non-uniform units (Taylor and Black, 1999; Segi et al., 2004).[2]

In all, there are many parameters to choose from by varying the size of the units, varying the size of the databases, and limiting the synthesis domain. Black highlighted these different directions in constructing the best unit-selection synthesizer for the targeted application (Black, 2002).

The mantra of "more data" may seem like an easy direction to follow, but with databases growing to tens of hours of data, time-dependent voice-quality variations have become a serious issue (Stylianou, 1999; Kawai and Tsuzaki, 2002; Shi et al., 2002). Also, very large databases require substantial computing resources that limit unit-selection techniques in embedded devices or where multiple voices and multiple languages are required.

These apparent issues specific to unit-selection synthesis are mentioned here because they have specific counterparts in statistical parametric synthesis.

3. Statistical parametric synthesis

3.1. Core architecture of typical system

In direct contrast to this selection of actual instances of speech from a database, statistical parametric speech synthesis might be most simply described as generating the *average* of some sets of similarly sounding speech segments. This contrasts directly with the need in unit-selection synthesis to retain natural unmodified speech units, but using parametric models offers other benefits.

In a typical statistical parametric speech synthesis system, we first extract parametric representations of speech including spectral and excitation parameters from a speech database and then model them by using a set of generative models (e.g., HMMs). A maximum likelihood (ML) criterion is usually used to estimate the model parameters as

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \{p(\boldsymbol{O}|\mathscr{W}, \boldsymbol{\lambda})\}, \tag{5}$$

where $\boldsymbol{\lambda}$ is a set of model parameters, $\boldsymbol{O}$ is a set of training data, and $\mathscr{W}$ is a set of word sequences corresponding to $\boldsymbol{O}$. We then generate speech parameters, $\boldsymbol{o}$, for a given word sequence to be synthesized, w, from the set of estimated models, $\hat{\boldsymbol{\lambda}}$, to maximize their output probabilities as

$$\hat{\boldsymbol{o}} = \arg\max_{\boldsymbol{o}} \left\{p(\boldsymbol{o}|w, \hat{\boldsymbol{\lambda}})\right\}. \tag{6}$$

Finally, a speech waveform is reconstructed from the parametric representations of speech. Although any generative model can be used, HMMs have been widely used. Statistical parametric speech synthesis with HMMs is commonly known as HMM-based speech synthesis (Yoshimura et al., 1999).

Fig. 3 is a block diagram of an HMM-based speech synthesis system. It consists of parts for training and synthesis. The training part performs the maximum likelihood estimation of Eq. (5) by using the EM algorithm (Dempster et al., 1977). This process is very similar to the one used for speech recognition, the main difference being that both spectrum (e.g., mel-cepstral coefficients (Fukada et al., 1992) and their dynamic features) and excitation (e.g., $\log F_0$ and its dynamic features) parameters are extracted from a database of natural speech and modeled by a set

[2] Note that a zero-cost join results from maintaining connectivity of units drawn from a unit-selection database and that implicitly yields a non-uniform unit-selection synthesizer.

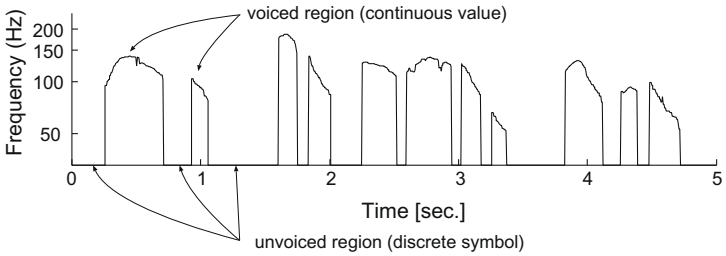

Fig. 4. Example of F_0 sequence with voiced and unvoiced regions.

of multi-stream (Young et al., 2006) context-dependent HMMs. Another difference is that linguistic and prosodic contexts are taken into account in addition to phonetic ones. For example, the contexts used in the HTS English recipes (Tokuda et al., 2008) are

- Phoneme:
  - –current phoneme,
  - –preceding and succeeding two phonemes,
  - –position of current phoneme within current syllable.
- Syllable:
  - –numbers of phonemes within preceding, current, and succeeding syllables,
  - –stress[3] and accent[4] of preceding, current, and succeeding syllables,
  - –positions of current syllable within current word and phrase,
  - –numbers of preceding and succeeding stressed syllables within current phrase,
  - –numbers of preceding and succeeding accented syllables within current phrase,
  - –number of syllables from previous stressed syllable,
  - –number of syllables to next stressed syllable,
  - –number of syllables from previous accented syllable,
  - –number of syllables to next accented syllable,
  - –vowel identity within current syllable.
- Word:
  - –guess at part of speech of preceding, current, and succeeding words,
  - –numbers of syllables within preceding, current, and succeeding words,
  - –position of current word within current phrase,
  - –numbers of preceding and succeeding content words within current phrase,
  - –number of words from previous content word,
  - –number of words to next content word.
- Phrase:
  - –numbers of syllables within preceding, current, and succeeding phrases,
  - –position of current phrase in major phrases,
  - –ToBI endtone of current phrase.
- Utterance:
  - –numbers of syllables, words, and phrases in utterance.

To model fixed-dimensional parameter sequences, such as mel-cepstral coefficients, single multi-variate Gaussian distributions are typically used as their stream-output distributions. However, it is difficult to apply discrete or continuous distributions to model variable-dimensional parameter sequences, such as $\log F_0$ sequences with unvoiced regions (Fig. 4). Although several methods have been investigated for modeling $\log F_0$ sequences (Freij and Fallside, 1988; Jensen et al., 1994; Ross and Ostendorf, 1994), the HMM-based speech synthesis system adopts multi-space probability distributions (Tokuda et al., 2002a) for its stream-output distributions.[5] Each HMM also has its state-duration distribution to model the temporal structure of speech (Yoshimura et al., 1998). Choices for state-duration distributions are the Gaussian distribution (Yoshimura et al., 1998) and the Gamma distribution (Ishimatsu et al., 2001). They are estimated from statistical variables obtained at the last iteration of the forward–backward algorithm. Each of spectrum, excitation, and duration is clustered individually by using decision trees (Odell, 1995) because they have their own context dependency. As a result, the system can model the spectrum, excitation, and duration in a unified framework.

The synthesis part performs the maximization of Eq. (6). This can be viewed as the inverse operation of speech recognition. First, a given word sequence is converted into a context-dependent label sequence, and then the utterance HMM is constructed by concatenating the context-dependent HMMs according to the label sequence. Second, the speech parameter generation algorithm generates the sequences of spectral and excitation parameters from the utterance HMM. Although there are several variants of the speech parameter generation algorithm (Tokuda et al., 2000; Tachiwa and Furui, 1999), the Case 1 algorithm in (Tokuda et al., 2000) has typically been used. Finally, a speech waveform is synthesized from the

[3] The lexical stress of the syllable as specified from the lexicon entry corresponding to the word related to this syllable.

[4] An intonational accent of the syllable predicted by a CART tree (0 or 1).

[5] Other F_0 modeling techniques such as Fujisaki's model (Hirose et al., 2005), quantification method type 1 (QMT1) (Iwano et al., 2002), and case-based reasoning (CBR) (Gonzalvo et al., 2007a) have also been used. Yu et al. also proposed a method of modeling $\log F_0$ sequences using standard HMMs (Yu et al., 2009).

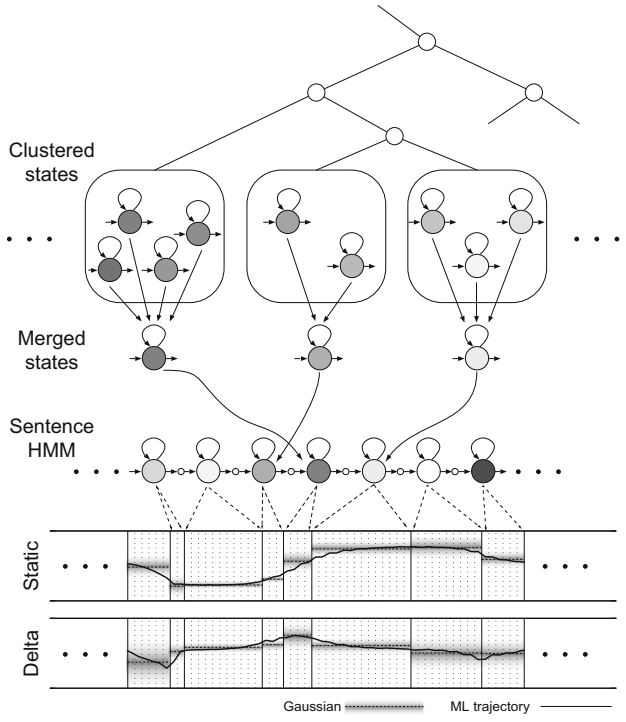

Fig. 5. Overview of HMM-based speech synthesis scheme.

generated spectral and excitation parameters using excitation generation and a speech synthesis filter (e.g., mel log spectrum approximation (MLSA) filter (Imai et al., 1983)). The following describes details of the speech parameter generation algorithm.

To simplify the notation here, we assume that each state-output distribution is a single stream, single multivariate Gaussian distribution as

$$b_j(\boldsymbol{o}_t) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j), \tag{7}$$

where $\boldsymbol{o}_t$ is the state-output vector at frame t, and $b_j(\cdot), \boldsymbol{\mu}_j$, and $\boldsymbol{\Sigma}_j$ correspond to the jth state-output distribution and its mean vector and covariance matrix. Within the HMM-based speech synthesis framework, Eq. (6) can be approximated as[6]

$$\hat{\boldsymbol{o}} = \arg\max_{\boldsymbol{o}} \left\{ p(\boldsymbol{o}|w, \hat{\boldsymbol{\lambda}}) \right\} \tag{8}$$

$$= \arg\max_{\boldsymbol{o}} \left\{ \sum_{\boldsymbol{q}} p(\boldsymbol{o}, \boldsymbol{q}|w, \hat{\boldsymbol{\lambda}}) \right\} \tag{9}$$

$$\approx \arg\max_{\boldsymbol{o}} \max_{\boldsymbol{q}} \left\{ p(\boldsymbol{o}, \boldsymbol{q}|w, \hat{\boldsymbol{\lambda}}) \right\} \tag{10}$$

$$= \arg\max_{\boldsymbol{o}} \max_{\boldsymbol{q}} \left\{ P(\boldsymbol{q}|w, \hat{\boldsymbol{\lambda}}) \cdot p(\boldsymbol{o}|\boldsymbol{q}, \hat{\boldsymbol{\lambda}}) \right\} \tag{11}$$

$$\approx \arg\max_{\boldsymbol{o}} \left\{ p(\boldsymbol{o}|\hat{\boldsymbol{q}}, \hat{\boldsymbol{\lambda}}) \right\} \tag{12}$$

$$= \arg\max_{\boldsymbol{o}} \left\{ \mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}_{\hat{\boldsymbol{q}}}, \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}}) \right\}, \tag{13}$$

where $\boldsymbol{o} = \left[\boldsymbol{o}_1^\top, \ldots, \boldsymbol{o}_T^\top\right]^\top$ is a state-output vector sequence to be generated, $\boldsymbol{q} = \{q_1, \ldots, q_T\}$ is a state sequence, and $\boldsymbol{\mu}_{\boldsymbol{q}} = \left[\boldsymbol{\mu}_{q_1}^\top, \ldots, \boldsymbol{\mu}_{q_T}^\top\right]^\top$ is the mean vector for $\boldsymbol{q}$. Here, $\boldsymbol{\Sigma}_{\boldsymbol{q}} = \mathrm{diag}\left[\boldsymbol{\Sigma}_{q_1}, \ldots, \boldsymbol{\Sigma}_{q_T}\right]$ is the covariance matrix for $\boldsymbol{q}$ and T is the total number of frames in $\boldsymbol{o}$. The state sequence, $\hat{\boldsymbol{q}}$, is determined to maximize its state-duration probability as

$$\hat{\boldsymbol{q}} = \arg\max_{\boldsymbol{q}} \left\{ P(\boldsymbol{q}|w, \hat{\boldsymbol{\lambda}}) \right\}. \tag{14}$$

Unfortunately, $\hat{\boldsymbol{o}}$ will be piece-wise stationary where the time segment corresponding to each state simply adopts the mean vector of the state. This would clearly be a poor fit to real speech where the variations in speech parameters are much smoother. To generate a realistic speech-parameter trajectory, the speech parameter generation algorithm introduces relationships between static and dynamic features as constraints for the maximization problem. If the state-output vector, $\boldsymbol{o}_t$, consists of the M-dimensional static feature, $\boldsymbol{c}_t$, and its first-order dynamic (delta) feature, $\Delta\boldsymbol{c}_t$, as

$$\boldsymbol{o}_t = \left[\boldsymbol{c}_t^\top, \Delta\boldsymbol{c}_t^\top\right]^\top \tag{15}$$

and the dynamic feature is calculated as[7]

$$\Delta\boldsymbol{c}_t = \boldsymbol{c}_t - \boldsymbol{c}_{t-1} \tag{16}$$

the relationship between $\boldsymbol{o}_t$ and $\boldsymbol{c}_t$ can be arranged in matrix form as

$$\begin{matrix} \boldsymbol{o} \\ \begin{bmatrix} \vdots \\ \boldsymbol{c}_{t-1} \\ \Delta\boldsymbol{c}_{t-1} \\ \boldsymbol{c}_t \\ \Delta\boldsymbol{c}_t \\ \boldsymbol{c}_{t+1} \\ \Delta\boldsymbol{c}_{t+1} \\ \vdots \end{bmatrix} \end{matrix} = \begin{matrix} \boldsymbol{W} \\ \begin{bmatrix} \cdots & \vdots & \vdots & \vdots & \vdots & \cdots \\ \cdots & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \cdots \\ \cdots & -\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \cdots \\ \cdots & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \cdots \\ \cdots & \boldsymbol{0} & -\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{0} & \cdots \\ \cdots & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \cdots \\ \cdots & \boldsymbol{0} & \boldsymbol{0} & -\boldsymbol{I} & \boldsymbol{I} & \cdots \\ \cdots & \vdots & \vdots & \vdots & \vdots & \cdots \end{bmatrix} \end{matrix} \begin{matrix} \boldsymbol{c} \\ \begin{bmatrix} \vdots \\ \boldsymbol{c}_{t-2} \\ \boldsymbol{c}_{t-1} \\ \boldsymbol{c}_t \\ \boldsymbol{c}_{t+1} \\ \vdots \end{bmatrix} \end{matrix} \tag{17}$$

where $\boldsymbol{c} = \left[\boldsymbol{c}_1^\top, \ldots, \boldsymbol{c}_T^\top\right]^\top$ is a static feature-vector sequence and $\boldsymbol{W}$ is a matrix that appends dynamic features to $\boldsymbol{c}$. Here, $\boldsymbol{I}$ and $\boldsymbol{0}$ correspond to the identity and zero matrices. As you can see, the state-output vectors are thus a linear transform of the static features. Therefore, maximizing $\mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}_{\hat{\boldsymbol{q}}}, \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}})$ with respect to $\boldsymbol{o}$ is equivalent to that with respect to $\boldsymbol{c}$:

$$\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{c}} \left\{ \mathcal{N}\left(\boldsymbol{W}\boldsymbol{c}; \boldsymbol{\mu}_{\hat{\boldsymbol{q}}}, \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}}\right) \right\}. \tag{18}$$

By equating $\partial \log \mathcal{N}(\boldsymbol{W}\boldsymbol{c}; \boldsymbol{\mu}_{\hat{\boldsymbol{q}}}, \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}})/\partial\boldsymbol{c}$ to $\boldsymbol{0}$, we can obtain a set of linear equations to determine $\hat{\boldsymbol{c}}$ as

$$\boldsymbol{W}^\top \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}}^{-1} \boldsymbol{W} \hat{\boldsymbol{c}} = \boldsymbol{W}^\top \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}}^{-1} \boldsymbol{\mu}_{\hat{\boldsymbol{q}}}. \tag{19}$$

Because $\boldsymbol{W}^\top \boldsymbol{\Sigma}_{\hat{\boldsymbol{q}}}^{-1} \boldsymbol{W}$ has a positive-definite band-symmetric structure, we can solve it very efficiently. The trajectory

[6] The Case 2 and 3 algorithms in (Tokuda et al., 2000), respectively, maximize Eqs. (9) and (8) under constraints between static and dynamic features.

[7] In the HTS English recipes (Tokuda et al., 2008), second-order (delta–delta) dynamic features are also used. The dynamic features are calculated as $\Delta\boldsymbol{c}_t = 0.5(\boldsymbol{c}_{t+1} - \boldsymbol{c}_{t-1})$ and $\Delta^2\boldsymbol{c}_t = \boldsymbol{c}_{t-1} - 2\boldsymbol{c}_t + \boldsymbol{c}_{t+1}$.

of $\hat{\boldsymbol{c}}$ will no longer be piece-wise stationary since associated dynamic features also contribute to the likelihood and therefore must be consistent with HMM parameters. Fig. 5 illustrates the effect of dynamic feature constraints. As we can see, the trajectory of $\hat{\boldsymbol{c}}$ becomes smooth rather than piece-wise.

3.2. Advantages

Most of the advantages of statistical parametric synthesis against unit-selection synthesis are related to its flexibility due to the statistical modeling process. The following describes details of these advantages.

3.2.1. Transforming voice characteristics, speaking styles, and emotions

The main advantage of statistical parametric synthesis is its flexibility in changing its voice characteristics, speaking styles, and emotions. Although the combination of unit-selection and voice conversion (VC) techniques (Stylianou et al., 1998) can alleviate this problem, high-quality voice-conversion is still problematic. Furthermore, converting prosodic features is also difficult. However, we can easily change voice characteristics, speaking styles, and emotions in statistical parametric synthesis by transforming its model parameters. There have been four major techniques to accomplish this, i.e., adaptation, interpolation, eigenvoice, and multiple regression.

3.2.1.1. Adaptation (mimicking voices). Techniques of adaptation were originally developed in speech recognition to adjust general acoustic models to a specific speaker or environment to improve the recognition accuracy (Leggetter and Woodland, 1995; Gauvain and Lee, 1994). These techniques have also been applied to HMM-based speech synthesis to obtain speaker-specific synthesis systems with a small amount of speech data (Masuko et al., 1997; Tamura et al., 2001). Two major techniques in adaptation are maximum a posteriori (MAP) estimation (Gauvain and Lee, 1994) and maximum likelihood linear regression (MLLR) (Leggetter and Woodland, 1995).

MAP estimation involves the use of prior knowledge about the distributions of model parameters. Hence, if we know what the parameters of the model are likely to be (before observing any adaptation data) using prior knowledge, we might well be able to make good use of the limited amount of adaptation data. The MAP estimate of an HMM, $\boldsymbol{\lambda}$, is defined as the mode of the posterior distribution of $\boldsymbol{\lambda}$, i.e.,

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \{p(\boldsymbol{\lambda}|\boldsymbol{O}, \mathscr{W})\} \tag{20}$$

$$= \arg\max_{\boldsymbol{\lambda}} \{p(\boldsymbol{O}, \boldsymbol{\lambda}|\mathscr{W})\} \tag{21}$$

$$= \arg\max_{\boldsymbol{\lambda}} \{p(\boldsymbol{O}|\mathscr{W}, \boldsymbol{\lambda}) \cdot p(\boldsymbol{\lambda})\}, \tag{22}$$

where $p(\boldsymbol{\lambda})$ is the prior distribution of $\boldsymbol{\lambda}$. A major drawback of MAP estimation is that every Gaussian distribution is individually updated. If the adaptation data is sparse, then many of the model parameters will not be updated. This causes the speaker characteristics of synthesized speech to often switch between general and target speakers within an utterance. Various attempts have been made to overcome this, such as vector field smoothing (VFS) (Takahashi and Sagayama, 1995) and structured MAP estimation (Shinoda and Lee, 2001).

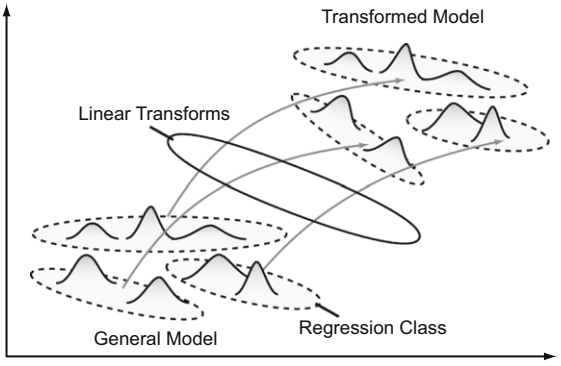

Fig. 6. Overview of linear-transformation-based adaptation technique.

Adaptation can also be accomplished by using MLLR and Fig. 6 gives an overview of this. In MLLR, a set of linear transforms is used to map an existing model set into a new adapted model set such that the likelihood for adaptation data is maximized. The state-output distributions[8] of the adapted model set are obtained as

$$b_j(\boldsymbol{o}_t) = \mathscr{N}(\boldsymbol{o}_t; \hat{\boldsymbol{\mu}}_j, \hat{\boldsymbol{\Sigma}}_j), \tag{23}$$

$$\hat{\boldsymbol{\mu}}_j = \boldsymbol{A}_{r(j)}\boldsymbol{\mu}_j + \boldsymbol{b}_{r(j)}, \tag{24}$$

$$\hat{\boldsymbol{\Sigma}}_j = \boldsymbol{H}_{r(j)}\boldsymbol{\Sigma}_j\boldsymbol{H}_{r(j)}^{\top}, \tag{25}$$

where $\hat{\boldsymbol{\mu}}_j$ and $\hat{\boldsymbol{\Sigma}}_j$ correspond to the linearly transformed mean vector and covariance matrix of the jth state-output distribution, and $\boldsymbol{A}_{r(j)}$, $\boldsymbol{H}_{r(j)}$, and $\boldsymbol{b}_{r(j)}$ correspond to the mean linear-transformation matrix, the covariance linear-transformation matrix, and the mean bias vector for the $r(j)$th regression class. The state-output distributions are usually clustered by a regression-class tree, and transformation matrices and bias vectors are shared among state-output distributions clustered into the same regression class (Gales, 1996). By changing the size of the regression-class tree according to the amount of adaptation data, we can control the complexity and generalization abilities of adaptation. There are two main variants of MLLR. If the same transforms are trained for $\boldsymbol{A}$ and $\boldsymbol{H}$, this is called constrained MLLR (or feature-space MLLR); otherwise, it is called unconstrained MLLR (Gales, 1998). For cases where adaptation data is limited, MLLR is currently a more effective form of adaptation than MAP estimation. Furthermore, MLLR offers adaptive training (Anastasakos et al., 1996; Gales, 1998), which can be used to estimate "canonical" models for training general models. For each

[8] The state-duration distributions can also be adapted in the same manner (Yamagishi and Kobayashi, 2007).

training speaker, a set of MLLR transforms is estimated, and then the canonical model is estimated given all these speaker transforms. Yamagishi applied this MLLR-based adaptive training and adaptation techniques to HMM-based speech synthesis (Yamagishi, 2006). This approach is called average voice-based speech synthesis (AVSS). It could be used to synthesize high-quality speech with the speaker's voice characteristics by only using a few minutes of the target speaker's speech data (Yamagishi et al., 2008b). Furthermore, even if hours of the target speaker's speech data were used, AVSS could still synthesize speech that had equal or better quality than speaker-dependent systems (Yamagishi et al., 2008c). Estimating linear-transformation matrices based on the MAP criterion (Yamagishi et al., 2009) and combining MAP estimation and MLLR have also been proposed (Ogata et al., 2006).

The use of the adaptation technique to create new voices makes statistical parametric speech synthesis more attractive. Usually, supervised adaptation is undertaken in speech synthesis, i.e., correct context-dependent labels that are transcribed manually or annotated automatically from texts and audio files are used for adaptation. As described in Section 3.1, phonetic, prosodic and linguistic contexts are used in speech synthesis. The use of such rich contexts makes unsupervised adaptation very difficult because generating context-dependent labels through speech recognition is computationally infeasible and likely to produce very inaccurate labels. King et al. proposed a simple but interesting solution to this problem by only using phonetic labels for adaptation (King et al., 2008). King et al. evaluated the performance of this approach and reported that the use of unsupervised adaptation degraded its intelligibility but its similarity to the target speaker and naturalness of synthesized speech were less severely impacted.

3.2.1.2. Interpolation (mixing voices). The interpolation technique enables us to synthesize speech with untrained voice characteristics. The idea of using interpolation was first applied to voice conversion, where pre-stored spectral patterns were interpolated among multiple speakers (Iwahashi and Sagisaka, 1995). It was also applied to HMM-based speech synthesis, where HMM parameters were interpolated among some representative HMM sets (Yoshimura et al., 1997). The main difference between Iwahashi and Sagisaka's technique and Yoshimura et al.'s one was that as each speech unit was modeled by an HMM, mathematically-well-defined statistical measures could be used to interpolate the HMMs. Fig. 7 illustrates the idea underlying the interpolation technique, whereby we can synthesize speech with various voice characteristics (Yoshimura et al., 1997), speaking styles (Tachibana et al., 2005), and emotions not included in the training data.

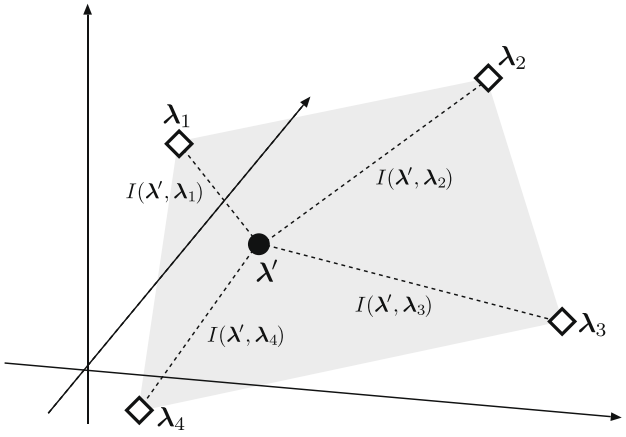

Fig. 7. Space of speaker individuality modeled by HMM sets $\{\boldsymbol{\lambda}_i\}$. In this figure, $\{I(\boldsymbol{\lambda}', \boldsymbol{\lambda}_i)\}$ denotes interpolation ratio.

3.2.1.3. Eigenvoice (producing voices). Although we can mimic voice characteristics, speaking styles, or emotions by only using a few utterances with the adaptation technique, we cannot obtain adapted models if no adaptation data is available. The use of the interpolation technique enables us to obtain various new voices by changing the interpolation ratio between representative HMM sets even if no adaptation data is available. However, if we increase the number of representative HMM sets to enhance the capabilities of representation, it is difficult to determine the interpolation ratio to obtain the required voice. To address this problem, Shichiri et al. applied the eigenvoice technique (Kuhn et al., 2000) to HMM-based speech synthesis (Shichiri et al., 2002). A speaker-specific "super-vector" was composed by concatenating the mean vectors of all state-output distributions in the model set for each S speaker-dependent HMM set. By applying principal component analysis (PCA) to S super-vectors $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_S\}$, we obtain eigen-vectors and eigen-values. By retaining lower-order eigen-vectors (larger eigen-values) and ignoring higher-order ones (small eigen-values), we can efficiently reduce the dimensionality of the speaker space because low-order eigen-vectors often contain the dominant aspects of given data. Using the first K eigen-vectors with arbitrary weights, we can obtain a new super-vector that represents a new voice as

$$\boldsymbol{s}' = \bar{\boldsymbol{\mu}} + \sum_{i=1}^{K} v_i' \boldsymbol{e}_i, \quad K < S, \tag{26}$$

where $\boldsymbol{s}'$ is a new super-vector, $\bar{\boldsymbol{\mu}}$ is a mean of the super-vectors, $\boldsymbol{e}_i$ is the ith eigen-vector, and v_i' is the weight for the ith eigen-vector. Then, a new HMM set can be reconstructed from $\boldsymbol{s}'$. Fig. 8 has an overview of the eigenvoice technique, which can reduce the number of parameters to be controlled, and this enables us to manually control the voice characteristics of synthesized speech by setting the weights. However, it introduces another problem in that it is difficult to control the voice characteristics intuitively because none of the eigen-vectors typically represent a specific physical meaning.

3.2.1.4. Multiple regression (controlling voices). To solve this problem, Miyanaga et al. applied a multiple-regression

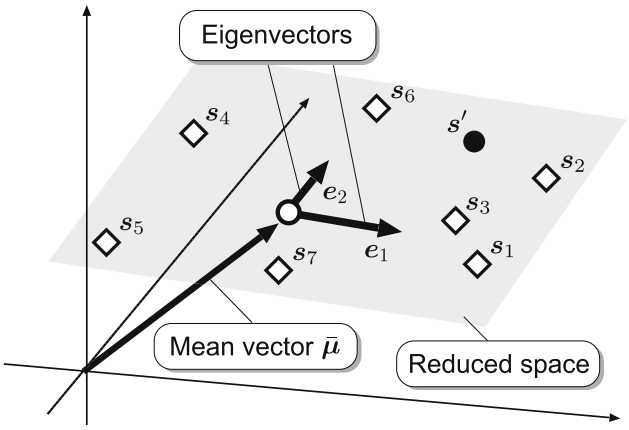

Fig. 8. Space of speaker individuality represented by super-vectors of HMM sets.

approach (Fujinaga et al., 2001) to HMM-based speech synthesis to control voice characteristics intuitively (Miyanaga et al., 2004; Nose et al., 2007b), where mean vectors of state-output distributions[9] were controlled with an L-dimensional control vector, $\boldsymbol{z} = [z_1, \ldots, z_L]^\top$, as

$$\boldsymbol{\mu}_j = \boldsymbol{M}_j \boldsymbol{\xi}, \quad \boldsymbol{\xi} = \left[1, \boldsymbol{z}^\top\right]^\top, \tag{27}$$

where $\boldsymbol{M}_j$ is a multiple-regression matrix. We can estimate $\{\boldsymbol{M}_j\}$ to maximize the likelihood of the model for the training data.[10] Each element of $\boldsymbol{z}$ captures specific voice characteristics, speaking styles, or emotions described by expressive words such as gender, age, brightness, and emotions, which are manually assigned through subjective listening tests. We can create any voices required in synthetic speech by specifying the control vector representing a point in a voice-characteristics space where each coordinate represents a specific characteristic. Estimating voice characteristics, speaking styles, and emotions of speech based on the multiple-regression technique has also been proposed (Nose et al., 2007a). Fig. 9 illustrates the idea underlying the multiple-regression technique, whereby we can intuitively control emotions in synthetic speech.

By combining these techniques, we can synthesize speech with various voice characteristics, speaking styles, and emotions without having to record large speech databases. For example, Tachibana et al. and Nose et al. proposed the combination of multiple-regression and adaptation techniques to achieve a multiple-regression technique with a small amount of speech data (Tachibana et al., 2008; Nose et al., 2009).

[9] The state-duration distributions can also be controlled in the same manner (Nose et al., 2007b).

[10] This training can be viewed as a special case of cluster adaptive training (CAT) (Gales, 2000), i.e., CAT estimates both $\boldsymbol{z}$ and $\boldsymbol{M}_j$ based on the ML criterion but the multiple-regression technique only estimates $\boldsymbol{M}_j$ and uses the provided control vector, $\boldsymbol{z}$, to assign an intuitive meaning to each cluster.

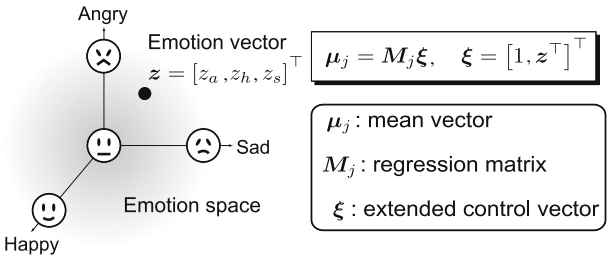

Fig. 9. Overview of multiple-regression HMM-based emotion-controlling technique.

3.2.2. Coverage of acoustic space

Unit-selection systems typically select from a *finite* set of units in the database. They search for the best path throughout a given set of units. Of course, when there are no good examples of units in that set, this can be viewed as either a lack of database coverage or that the required sentence to be synthesized is not in the domain. To alleviate this problem, many systems do some localized smoothing at segment boundaries. Wouters and Macon and Tamura et al. introduced the notion of *fusion units* effectively increasing the number of available units by allowing new units to be constructed from existing ones (Wouters and Macon, 2000; Tamura et al., 2005). In contrast to unit-selection synthesis, statistical parametric synthesis uses statistics to *generate* speech. Thus, a much wider range of units is effectively available, as context affects the generation of speech parameters through constraining dynamic features, and smoother joins are possible. However, while it can potentially cover the given acoustic space better than unit-selection systems, it is still limited by the examples in the database.

3.2.3. Multilingual support

Supporting multiple languages can easily be accomplished in statistical parametric speech synthesis because only the contextual factors to be used depend on each language. Furthermore, we can create statistical parametric speech synthesis systems with a small amount of training data. Takamido et al. demonstrated that an intelligible HMM-based speech synthesis system could be built by using approximately 10 min from a single-speaker, phonetically balanced speech database. This property is important to support numerous languages because few speech and language resources are available in many languages. Note that the contexts to be used should be designed for each language to achieve better quality of synthesis because each language has its own contextual factors. For example, the use of tonal contexts is essential in tonal languages such as Mandarin Chinese. As of this point, Arabic (Abdel-Hamid et al., 2006; Fares et al., 2008), Basque, Catalan (Bonafonte et al., 2008), Mandarin Chinese (Zen et al., 2003a; Wu and Wang, 2006a; Qian et al., 2006), Croatian (Martincic-Ipsic and Ipsic, 2006), Dzongkha (Sherpa et al., 2008), US (Tokuda et al., 2002b), UK, Scottish,

Canadian, Indian, and South African English, Farsi (Homayounpour and Mehdi, 2004), Finnish (Ojala, 2006; Vainio et al., 2005; Raitio et al., 2008; Silen et al., 2008), French (Drugman et al., 2008), Scottish Gaelic (Berry, 2008), standard (Weiss et al., 2005; Krstulović et al., 2007), Austrian, Viennese sociolect and dialect German, Greek (Karabetsos et al., 2008), Hebrew, Hindi, Hungarian (Tóth and Németh, 2008), Indonesian (Sakti et al., 2008), Irish, Japanese (Yoshimura et al., 1999), Korean (Kim et al., 2006b), Lao, Mongolian, Polish, European (Barros et al., 2005) and Brazilian (Maia et al., 2003) Portuguese, Russian, Serbian, Slovak (Sýkora, 2006), Slovenian (Vesnicer and Mihelic, 2004), Spanish (Gonzalvo et al., 2007b), Swedish (Lundgren, 2005), Tamil, Telugu, Thai (Chomphan and Kobayashi, 2007), Vietnamese, Xhosa, Zulu, and Mandarin Chinese–English bilingual (Liang et al., 2008; Qian et al., 2008a) systems have been or are being built by various groups.

Speech synthesizers in new languages have typically been constructed by collecting several hours of well-recorded speech data in the target language. An alternative method has been to apply the same idea as in speech recognition, i.e., to use a multilingual acoustic model from an existing synthesizer in one language and cross adapt models to the target language based on a very small set of collected sentences. To utilize speech data from multiple speakers and multiple languages for speech synthesis, unit-selection synthesis is unlikely to succeed given that it has a wider variety of data and less consistency. However, within statistical parametric synthesis, the adaptive training and adaptation framework allows multiple speakers and even languages to be combined into single models, thus enabling multilingual synthesizers to be built. Latorre et al. and Black and Schultz proposed building such multilingual synthesizers using combined data from multiple languages (Latorre et al., 2006; Black and Schultz, 2006). Wu et al. also proposed a technique of cross-lingual speaker adaptation (Wu et al., 2008a). They revealed that multilingual synthesis and cross-lingual adaptation were indeed feasible and provided reasonable quality.

3.2.4. Other advantages

3.2.4.1. Footprint. When compared with unit-selection synthesis, the footprint of statistical parametric synthesis is usually small because we store statistics of acoustic models rather than the multi-templates of speech units. For example, the footprints of Nitech's Blizzard Challenge 2005 voices were less than 2 MBytes with no compression (Zen et al., 2007c). Their footprints could be further reduced without any degradation in quality by eliminating redundant information. Additional reduction was also possible with small degradation in quality by utilizing vector quantization, using fixed-point numbers instead of floating-point numbers, pruning decision trees (Morioka et al., 2004), and/or tying model parameters (Oura et al., 2008b). For example, Morioka et al. demonstrated that HMM-based speech synthesis systems whose footprints were about 100 KBytes could synthesize intelligible speech by properly tuning various parameters (Morioka et al., 2004). Taking these into consideration, we believe that statistical parametric speech synthesis seems to be suitable for embedded applications (Kim et al., 2006a). A memory-efficient, low-delay speech parameter generation algorithm (Tokuda et al., 1995; Koishida et al., 2001) and a computationally-efficient speech synthesis filter (Watanabe et al., 2007), which seem useful for incorporating HMM-based speech synthesis into embedded devices, have been proposed. Several commercial products based on statistical parametric speech synthesis for mobile devices have been released (SVOX AG, 2007; Bai, 2007; KDDI R&D, 2008; SVOX AG, 2008).

3.2.4.2. Robustness. Statistical parametric speech synthesis is more "robust" than unit-selection synthesis. If we want to build speech synthesizers using speech data from real users, the speech from the target speaker could possibly suffer from noise or fluctuations due to the recording conditions. This would be expected to significantly degrade the quality of synthetic speech. Furthermore, such data is unlikely to be phonetically balanced and therefore lack some units. Yamagishi et al. reported that statistical parametric speech synthesis, especially AVSS, was much more robust to these kinds of factors (Yamagishi et al., 2008a). This is because adaptive training can be viewed as a general version of several feature-normalization techniques such as cepstral mean/variance normalization, stochastic matching, and bias removal. Furthermore, the use of an average-voice model can provide supplementary information that is lacking in the adaptation data. They also reported that "recording condition-adaptive training," which is based on the same idea as speaker-adaptive training (Anastasakos et al., 1996; Gales, 1998), worked effectively to normalize recording conditions.

3.2.4.3. Using speech recognition technologies. Statistical parametric speech synthesis, especially HMM-based speech synthesis, can employ a number of useful technologies developed for HMM-based speech recognition. For example, structured precision matrix models (Gales, 1999; Olsen and Gopinath, 2004), which can closely approximate full covariance models using small numbers of parameters, have successfully been applied to a system (Zen et al., 2006b).

3.2.4.4. Unifying front-end and back-end. Statistical parametric speech synthesis provides a new framework for jointly optimizing the front-end (text analysis) and back-end (waveform generation) modules of text-to-speech (TTS) systems. These two modules are conventionally constructed independently. The text-analysis module is trained using text corpora and often includes statistical models to analyze text, e.g., the phrasing boundary, accent, and POS. The waveform generation module, on the other hand, is trained using a labeled speech database. In statistical

parametric synthesis, this module includes acoustic models. If these two modules are jointly estimated as a unified statistical model, it is expected to improve the overall performance of a TTS system. Based on this idea, Oura et al. proposed an integrated model for linguistic and acoustic modeling and demonstrated its effectiveness (Oura et al., 2008a).

3.2.4.5. Fewer tuning parameters. Unit-selection synthesis usually requires various control parameters to be manually tuned. Statistical parametric synthesis, on the other hand, has few tuning parameters because all the modeling and synthesis processes are based on mathematically well-defined statistical principles.

3.2.4.6. Separately control spectrum, excitation, and duration. Because statistical parametric speech synthesis uses the source-filter representation of speech, the spectrum, excitation, and duration can be controlled and modified separately.

3.3. Drawbacks and refinements

The biggest drawback with statistical parametric synthesis versus unit-selection synthesis is the quality of synthesized speech. There seem to be three factors that degrade quality, i.e., vocoders, acoustic modeling accuracy, and over-smoothing. Details on these factors and various refinements that are needed to achieve state-of-the-art performance are described in the following.

3.3.1. Vocoder

The speech synthesized by the basic HMM-based speech synthesis system sounds *buzzy* since it uses a mel-cepstral vocoder with simple periodic pulse-train or white-noise excitation (Yoshimura et al., 1999).

3.3.1.1. Excitation model. To alleviate this problem, high-quality vocoders such as mixed excitation linear prediction (MELP) (Yoshimura et al., 2001; Gonzalvo et al., 2007b), multi-band excitation (Abdel-Hamid et al., 2006), pitch synchronous residual codebook (Drugman et al., 2009) the harmonic plus noise model (HNM) (Hemptinne, 2006; Kim and Hahn, 2007), the flexible pitch-asynchronous harmonic/stochastic model (HSM) (Banos et al., 2008), STRAIGHT (Zen et al., 2007c), the glottal-flow-derivative model (Cabral et al., 2007; Cabral et al., 2008), or the glottal waveform (Raitio et al., 2008) have been integrated. The most common feature in most of these methods is the fact that they are based on the implementation of an excitation model through the utilization of some *special parameters* modeled by HMMs; they do not directly minimize the distortion between artificial excitation and speech residuals.

Maia et al. have recently proposed a trainable technique of excitation modeling for HMM-based speech synthesis (Maia et al., 2007). Fig. 10 has a block diagram of this.

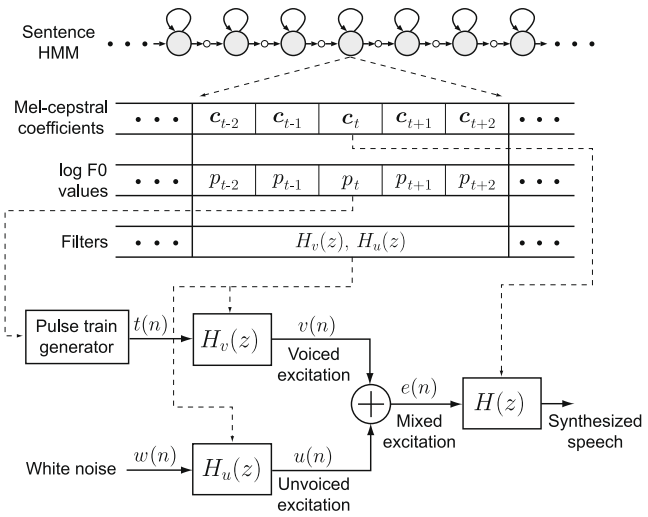

Fig. 10. ML-based excitation scheme proposed by Maia et al. for HMM-based speech synthesis: filters $H_v(z)$ and $H_u(z)$ are associated with each state.

In this technique, mixed excitation is produced by inputting periodic pulse trains and white noise into two state-dependent filters. These specific states can be built using bottom-up (Maia et al., 2008) or top-down (Maia et al., 2009) clustering method. The filters are derived to maximize the likelihood of residual sequences over corresponding states through an iterative process. Apart from determining the filter, the amplitudes and positions of the periodic pulse trains have also been optimized in the sense of residual likelihood maximization during referred closed-loop training. As a result, this technique directly minimizes the weighted distortion (Itakura-Saito distance (Itakura, 1975)) between the generated excitation and speech residual. This technique is very similar to the closed-loop training for unit-concatenation synthesis (Akamine and Kagoshima, 1998). Both of them are based on the idea of a code excitation linear prediction (CELP) vocoder. However, there is an essential difference between these two techniques. Maia et al.'s technique targets residual modeling but Akamine and Kagoshima's technique targets a one-pitch waveform. Furthermore, Maia et al.'s technique includes both voiced and unvoiced components for the waveform-generation part. Fig. 11 shows a transitional segment of natural speech and three types of synthesized speech obtained by natural spectra and F_0 with the simple periodic pulse-train or white-noise excitation, the STRAIGHT's excitation, and Maia et al.'s ML excitation modeling methods. The residual signal derived through inverse filtering of a natural speech signal and the corresponding excitation signals and synthesized speech are also shown. We can see that the method of ML excitation modeling produces excitation and speech waveforms that are closer to the natural ones.

3.3.1.2. Spectral representation of speech. Several groups have recently applied LSP-type parameters instead of

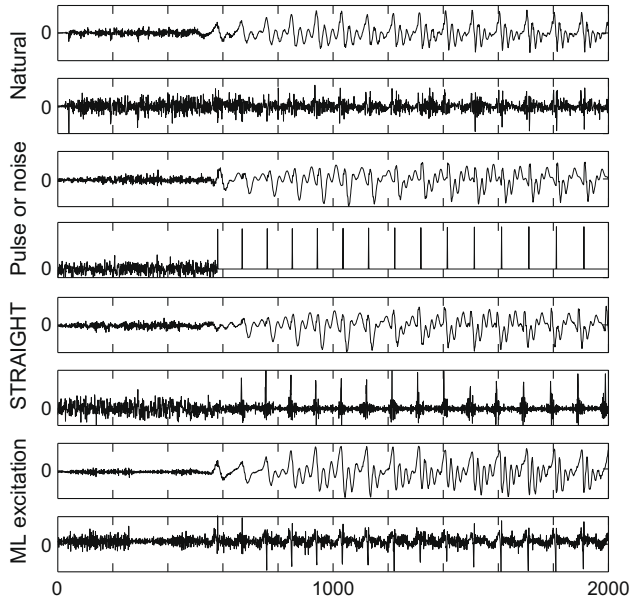

Fig. 11. Waveforms from top to bottom: natural speech and its residual, speech and excitation synthesized with simple periodic pulse-train or white-noise excitation, speech and excitation synthesized with STRAIGHT vocoding method, and speech and excitation synthesized with ML excitation method.

cepstral parameters to HMM-based speech synthesis (Nakatani et al., 2006; Ling et al., 2006; Zen et al., 2006b; Qian et al., 2006). As is well known, LSP-type parameters have good quantization and interpolation properties and have successfully been applied to speech coding. These characteristics seem to be valuable in statistical parametric synthesis because statistical modeling is closely related to quantization and synthesis is closely related to interpolation. Marume et al. compared LSPs, log area ratios (LARs), and cepstral parameters in HMM-based speech synthesis and reported that LSP-type parameters achieved the best subjective scores for these spectral parameters (Marume et al., 2006). Kim et al. also reported that 18th-order LSPs achieved almost the same quality as 24th-order mel-cepstral coefficients (Kim et al., 2006a).

Although LSP-type parameters have various advantages over cepstral ones, they also have drawbacks. It is well known that as long as the LSP coefficients are within 0–π and in ascending order the resulting synthesis filter will be stable. However, it is difficult to guarantee whether LSPs generated from HMMs will satisfy these properties because state-output distributions are usually Gaussian distributions with diagonal covariance matrices. This problem becomes more prominent when we transform model parameters (Qin et al., 2006). Although the use of a full covariance model or its approximations (Zen et al., 2006b), band constraints in linear-transformation matrices (Qin et al., 2006), or differentials of adjacent LSPs (Qian et al., 2006) can reduce the effect this problem incurs, we still cannot guarantee that the resulting synthesis filter will become stable. Combining spectral estimation and observation modeling would fundamentally be essential to solving this problem. Several techniques of combining spectral analysis and model training have recently been proposed. Acero integrated formant analysis (Acero, 1999), Toda and Tokuda incorporated cepstral analysis (Toda and Tokuda, 2008), and Wu and Tokuda combined LSP parameter extraction (Wu and Tokuda, 2009). These techniques, especially those of (Toda and Tokuda, 2008; Wu and Tokuda, 2009), are based on a similar concept to analysis-by-synthesis in speech coding and the closed-loop training (Akamine and Kagoshima, 1998) for concatenative speech synthesis. Such closed-loop training can eliminate the mismatch between spectral analysis, acoustic-model training, and speech-parameter generation, and thus improves the quality of synthesized speech. Signal process-embedded statistical models like auto-regressive HMMs (Penny et al., 1998) and frequency-warped exponential HMMs (Takahashi et al., 2001) may also be useful to solve this problem.

3.3.2. Accuracy of acoustic modeling

Hidden Markov models perform well considering the various postulations made in using them, such as piece-wise constant statistics within an HMM state, the assumption of frame-wise conditional independence of state-output probabilities, and simple geometric state-duration distributions. However, none of these assumptions hold for real speech. Because speech parameters are directly generated from acoustic models, their accuracy affects the quality of synthesized speech. We can expect that the use of a more precise statistical model will improve the quality of synthesized speech.

3.3.2.1. Better acoustic model. One way of increasing the accuracy of the acoustic model is using dynamic models that can capture the explicit dynamics of speech-parameter trajectories. To alleviate the problem with piece-wise constant statistics, Dines and Sridharan applied trended HMMs (Deng, 1992), which included linearly time-varying functions in their state-output probabilities, to statistical parametric synthesis (Dines and Sridharan, 2001). Similarly, Sun et al. used a polynomial segment model (Gish and Ng, 1993) to describe speech-parameter trajectories (Sun et al., 2009). Bulyko et al. introduced buried Markov models (Bilmes, 2003), which had additional dependencies between observation elements to increase accuracy, to statistical parametric synthesis (Bulyko et al., 2002) to avoid the assumption of conditional independence. These dynamic models were evaluated in small tasks and they were found to work slightly better than HMMs. However, HMMs are still being used as dominant acoustic models in statistical parametric synthesis because these dynamic models require the number of model parameters to be increased. Furthermore, various essential algorithms such as decision-tree-based context clustering (Odell, 1995) need to be re-derived for these dynamic models.

Zen et al. recently showed that an HMM whose state-output vector included both static and dynamic features

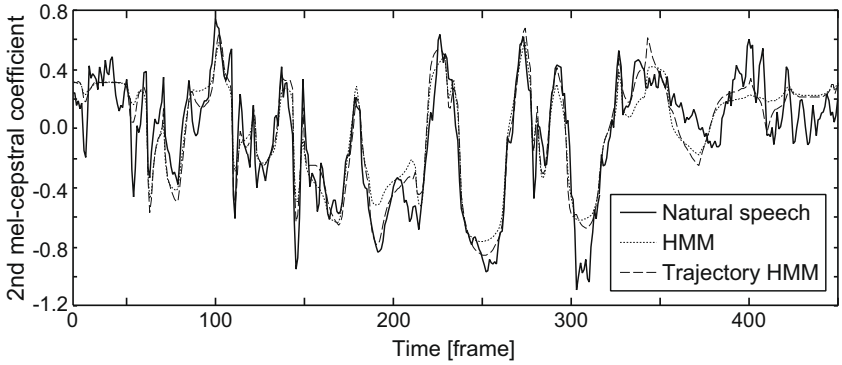

Fig. 12. Trajectories of second mel-cepstral coefficients of natural and synthesized speech generated from ML-estimated HMMs and trajectory HMMs. In this figure, solid, dashed, and dotted lines correspond to natural trajectory, that generated from HMMs, and that generated from trajectory HMMs.

could be reformulated as a trajectory model by imposing explicit relationships between static and dynamic features (Zen et al., 2006c). This model, called a trajectory HMM, could overcome the assumption of conditional independence and constant statistics within an HMM state without the need for any additional parameters. It is defined as

$$p(\boldsymbol{c}|\boldsymbol{\lambda}) = \sum_{\forall \boldsymbol{q}} P(\boldsymbol{q}|\boldsymbol{\lambda}) \cdot p(\boldsymbol{c}|\boldsymbol{q}, \boldsymbol{\lambda}), \tag{28}$$

$$p(\boldsymbol{c}|\boldsymbol{q}, \boldsymbol{\lambda}) = \mathcal{N}\left(\boldsymbol{c}; \bar{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}\right), \tag{29}$$

$$P(\boldsymbol{q}|\boldsymbol{\lambda}) = P(q_1|\boldsymbol{\lambda}) \prod_{t=2}^{T} P(q_t|q_{t-1}, \boldsymbol{\lambda}), \tag{30}$$

where $\bar{\boldsymbol{c}}_{\boldsymbol{q}}$ is the $MT \times 1$ mean vector for $\boldsymbol{q}$, $\boldsymbol{P}_{\boldsymbol{q}}$ is the $MT \times MT$ covariance matrix, M is the dimensionality of static features, and T is the total number of frames in $\boldsymbol{c}$. They are given by

$$\boldsymbol{R}_{\boldsymbol{q}} \bar{\boldsymbol{c}}_{\boldsymbol{q}} = \boldsymbol{r}_{\boldsymbol{q}}, \tag{31}$$

$$\boldsymbol{R}_{\boldsymbol{q}} = \boldsymbol{W}^{\top} \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \boldsymbol{W} = \boldsymbol{P}_{\boldsymbol{q}}^{-1}, \tag{32}$$

$$\boldsymbol{r}_{\boldsymbol{q}} = \boldsymbol{W}^{\top} \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \boldsymbol{\mu}_{\boldsymbol{q}}. \tag{33}$$

This model is closely related to the speech parameter generation algorithm (Tokuda et al., 2000) used in HMM-based speech synthesis; the mean vector of the trajectory HMM, $\bar{\boldsymbol{c}}_{\boldsymbol{q}}$, which is given by solving the set of linear equations in Eq. (31), is identical to the speech-parameter trajectory, $\hat{\boldsymbol{c}}$, which is given by solving the set of linear equations in Eq. (19). This is because both of these are derived from the HMM with explicit relationships between static and dynamic features. Hence, estimating trajectory HMMs based on the ML criterion, i.e.,

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \{p(\boldsymbol{C}|\mathcal{W}, \boldsymbol{\lambda})\}, \tag{34}$$

where $\boldsymbol{C}$ is a set of training data (static feature-vector sequences only), can be viewed as closed-loop training for HMM-based speech synthesis. Fig. 12 shows what effect trajectory HMM training has. We can see from the figure that the trajectory generated from the trajectory HMMs is closer to the training data than that from the HMMs. Similar work has been carried out by Wu and Wang (Wu and Wang, 2006b). They proposed minimum generation error (MGE) training[11] for HMM-based speech synthesis, which estimates model parameters to minimize the Euclidean distance[12] between training data and generated speech parameters as

$$\hat{\boldsymbol{\lambda}} = \arg\min_{\boldsymbol{\lambda}} \{\mathcal{E}(\boldsymbol{C}; \mathcal{W}, \boldsymbol{\lambda})\}, \tag{35}$$

where $\mathcal{E}(\boldsymbol{C}; \mathcal{W}, \boldsymbol{\lambda})$ is the expected total Euclidean distance between the training data and generated parameters. This is equivalent to estimating trajectory HMM parameters based on the minimum mean squared error (MMSE) criterion instead of that of ML (Zhang, 2009). It can also be viewed as estimating the following statistical model based on the ML criterion:

$$p(\boldsymbol{c}|\boldsymbol{\lambda}) = \sum_{\forall \boldsymbol{q}} P(\boldsymbol{q}|\boldsymbol{\lambda}) \cdot p(\boldsymbol{c}|\boldsymbol{q}, \boldsymbol{\lambda}), \tag{36}$$

$$p(\boldsymbol{c}|\boldsymbol{q}, \boldsymbol{\lambda}) = \mathcal{N}\left(\boldsymbol{c}; \bar{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{I}\right), \tag{37}$$

$$P(\boldsymbol{q}|\boldsymbol{\lambda}) = P(q_1|\boldsymbol{\lambda}) \prod_{t=2}^{T} P(q_t|q_{t-1}, \boldsymbol{\lambda}). \tag{38}$$

Eq. (35) was iteratively minimized in an on-line fashion using the generalized probabilistic decent (GPD) algorithm (Katagiri et al., 1991), which has been used for minimum classification error (MCE) training in speech recognition (Juang et al., 1997).[13] Both of these significantly improved the quality of synthesis over the conventional ML-estimated HMM on full systems. One of the advantages of trajectory HMM over other dynamic models is that huge amounts of software resources or algorithms developed for HMMs can easily be reused (Wu et al., 2006; Zen et al., 2006a; Zen et al., 2007a; Qin et al., 2008) because its parameterization is equivalent to that of HMMs.

[11] The term "Trajectory HMM" denotes the name of generative models like "HMM." However, the term "MGE" represents the name of parameter-optimization criteria like "ML" or "MCE."

[12] Minimizing the log spectral distance (MGE-LSD) between training data and generated LSP coefficients has also been proposed (Wu and Tokuda, 2008; Wu and Tokuda, 2009).

[13] Although the MGE training algorithm adopts the GPD-based optimization technique, its loss function is squared error. Therefore, MGE training is not discriminative training.

3.3.2.2. Better duration model. The state-duration probability in the jth state for d consecutive frames in an HMM is given by

$$p_j(d) = a_{jj}^{d-1}(1 - a_{jj}), \tag{39}$$

where a_{jj} is the self-transition probability of the j state. Fig. 13 plots an example of the state-duration probability of an HMM. We can see from the figure that the HMM models the state-duration probability as decreasing exponentially with time and this is clearly a poor model of duration. To overcome this problem, the HMM-based speech synthesis system models state-duration distributions explicitly with Gaussian (Yoshimura et al., 1998) or Gamma distributions (Ishimatsu et al., 2001). They are estimated from statistical variables obtained at the last iteration of the forward–backward algorithm, and then clustered by using a decision tree; they are not re-estimated in the Baum–Welch iteration. At the synthesis stage, we construct a sentence HMM and determine its state durations to maximize their probabilities. Then, speech parameters are generated. However, there are inconsistencies. Although the parameters of HMMs are estimated without explicit state-duration distributions, speech parameters are generated from HMMs using explicit state-duration distributions. These inconsistencies can degrade the quality of synthesized speech. To resolve this discrepancy, hidden semi-Markov models (Ferguson, 1980; Russell and Moore, 1985; Levinson, 1986), which can be viewed as HMMs with explicit state-duration distributions, were introduced (Zen et al., 2007d). The use of HSMMs makes it possible to simultaneously re-estimate state-output and state-duration distributions. Adaptation and adaptive training techniques for HSMMs have also been derived (Yamagishi and Kobayashi, 2007). Although improvements in speaker-dependent systems have been small (Zen et al., 2007d), it is essential to adapt state-duration distributions (Tachibana et al., 2006). Multi-level duration models including phoneme- and syllable-duration distributions in addition to state-duration distributions have also been proposed to achieve better duration modeling (Wu and Wang, 2006a; Gao et al., 2008). Lu et al. investigated the use of full covariance Gaussian distributions to model the inter-state correlations of state durations (Lu et al., 2009).

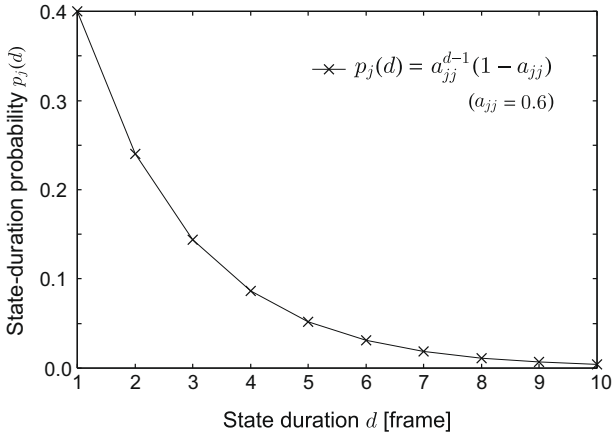

Fig. 13. Example of state-duration probability of HMM ($a_{jj} = 0.6$).

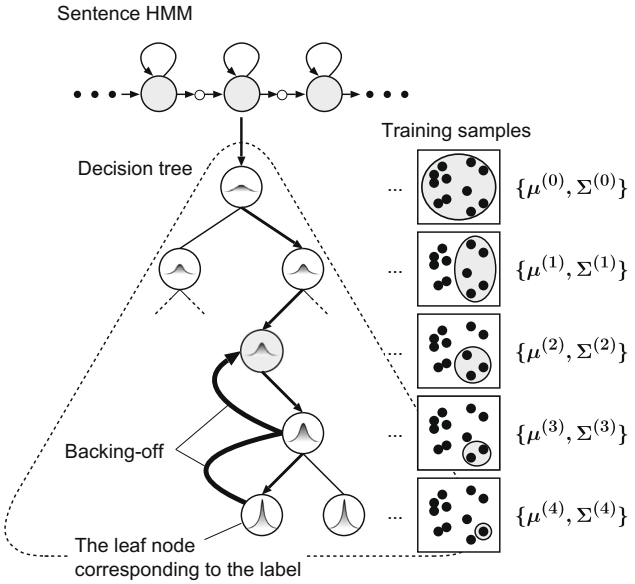

Fig. 14. Overview of decision-tree back-off technique for HMM-based speech synthesis.

3.3.2.3. Complexity control. Another way of enhancing the accuracy of models is by increasing the number of parameters. However, using too many parameters results in overfitting these to the training data. We can reproduce training sentences with excellent quality but may synthesize unseen sentences with poor quality. Therefore, controlling model complexity while retaining its capability for generalization is very important to achieve superior synthesized speech. Minimum description length (MDL) (Rissanen, 1980) criterion-based decision-tree clustering (Shinoda and Watanabe, 2000) has been used in the HMM-based speech synthesis system to balance model complexity and accuracy. However, since the MDL criterion is derived based on an asymptotic assumption, it is theoretically invalid when there is little training data because the assumption fails. This situation often occurs in speech synthesis because the amount of training data used in speech synthesis is often much smaller than the amount used in speech recognition. Furthermore, Watanabe found that the MDL criterion[14] could not be applied to statistical models that included hidden variables (Watanabe, 2007).

One possible solution to this problem is dynamically changing the complexity of models. Kataoka et al. proposed a decision-tree backing-off technique for HMM-based speech synthesis (Kataoka et al., 2004) as shown in Fig. 14. It could dynamically vary the size of phonetic decision trees at run-time according to the text to be

[14] Also, neither the Bayesian information criterion (BIC) (Schwarz, 1978) nor Akaike's information criterion (AIC) (Akaike, 1974) can be applied.

synthesized. Similarly, unit-selection synthesis systems using backing-off methods have also been proposed (e.g., Donovan and Eide, 1998). However, Kataoka's technique differs from these because backing-off is undertaken to maximize the output probability of speech-parameter trajectories.

Another possible solution is using the Bayesian-learning framework. Bayesian learning is used to estimate the posterior distributions of model parameters from prior distributions and training data whereas ML and MAP learning are used to estimate the parameter values (point estimates). This property enables us to incorporate prior knowledge into the estimation process and improves model generalization due to the marginalization effect of model parameters. It offers selection of model complexity in the sense of maximizing its posterior probability. Recently, Watanabe et al. applied the variational Bayesian-learning technique (Beal, 2003) to speech recognition (Watanabe et al., 2004), and Nankaku et al. applied this idea to HMM-based speech synthesis (Nankaku et al., 2003). Bayesian statistical parametric synthesis determines $\boldsymbol{o}$ as

$$\hat{\boldsymbol{o}} = \arg\max_{\boldsymbol{o}} \{p(\boldsymbol{o}|w, \boldsymbol{O}, \mathscr{W})\} \tag{40}$$

$$= \arg\max_{\boldsymbol{o}} \{p(\boldsymbol{o}, \boldsymbol{O}|w, \mathscr{W})\} \tag{41}$$

$$= \arg\max_{\boldsymbol{o}} \left\{ \int p(\boldsymbol{o}, \boldsymbol{O}, \boldsymbol{\lambda}|w, \mathscr{W})\, d\boldsymbol{\lambda} \right\} \tag{42}$$

$$= \arg\max_{\boldsymbol{o}} \left\{ \int p(\boldsymbol{o}, \boldsymbol{O}|w, \mathscr{W}, \boldsymbol{\lambda}) \cdot p(\boldsymbol{\lambda})\, d\boldsymbol{\lambda} \right\} \tag{43}$$

$$= \arg\max_{\boldsymbol{o}} \left\{ \int p(\boldsymbol{o}|w, \boldsymbol{\lambda}) \cdot p(\boldsymbol{O}|\mathscr{W}, \boldsymbol{\lambda}) \cdot p(\boldsymbol{\lambda})\, d\boldsymbol{\lambda} \right\}, \tag{44}$$

where $p(\boldsymbol{o}|w, \boldsymbol{O}, \mathscr{W})$ is the predictive distribution of $\boldsymbol{o}$.

Eq. (40) is the fundamental problem that needs to be solved in corpus-based speech synthesis, i.e., finding the most likely speech parameters, $\hat{\boldsymbol{o}}$, for a given word sequence, w, using the training data, $\boldsymbol{O}$, and the corresponding word sequence, $\mathscr{W}$. The equations above also indicate that $\boldsymbol{o}$ is generated from the predictive distribution, which is analytically derived from the marginalization of $\boldsymbol{\lambda}$ based on the posterior distribution estimated from $\boldsymbol{O}$. We can solve this maximization problem by using Bayesian speech parameter generation algorithms (Nankaku et al., 2003), which are similar to the ML-based speech parameter generation algorithms (Tokuda et al., 2000). One research topic in the Bayesian approach is how to set the hyper-parameters[15] of the prior distribution, because the quality of synthesized speech is sensitive to these. These hyper-parameters have been set empirically in the conventional approaches. Hashimoto et al. recently proposed a cross-validation (CV)-based technique of setting hyper-parameters (Hashimoto et al., 2008) for Bayesian speech synthesis. It demonstrated that the CV-based Bayesian speech synthesizer achieved better quality synthesized speech than an ML-based one.

[15] A hyper-parameter is a parameter of the prior distribution.

3.3.2.4. Model topology. Another research topic in acoustic modeling is *model topology*. A three or five-state left-to-right structure is used for all phonemes in the HMM-based speech synthesis system. This is apparently unsuitable because all phonemes have different durations and co-articulations. Related to this topic, Eichner et al. applied stochastic Markov graphs, which have enhanced capabilities for modeling trajectories, to statistical parametric synthesis (Eichner et al., 2000). Although this offers a flexible topology, it requires a search process for the state sequence at the synthesis stage (Eichner et al., 2001) because we need to determine a single-state sequence to generate speech parameters efficiently using the speech parameter generation algorithm (the Case 1 algorithm in (Tokuda et al., 2000)). Although we can skip this process by marginalizing all possible state sequences using an EM-type parameter generation algorithm (the Case 3 algorithm in (Tokuda et al., 2000)), this further increases computational complexity. Taylor also discussed the model topology of HMM-based speech synthesis (Taylor, 2006). The details will be given in Section 4.2.4.

The parameter-tying level can also be viewed as model-topology research. In the HMM-based speech synthesis system, a stream-level-tying structure has been used, i.e., spectral-parameter and excitation-parameter streams are individually clustered by decision trees. Yu et al. investigated more local-level-tying structures (e.g., splitting 0th cepstral coefficients and splitting static and dynamic features) for modeling spectral features (Yu et al., 2008). Similarly, Shinta et al. also investigated the use of a feature-dependent tying structure with asynchronous transition (Matsuda et al., 2003) in HMM-based speech synthesis (Shinta et al., 2005). They revealed that some local-level-tying structures worked better than the stream-level-tying structure but that excessively aggressive separation was not feasible. The dimension-split technique such as (Zen et al., 2003b) is expected to be useful for automatically determining the local-level-tying structure.

Most statistical parametric speech synthesis systems directly model acoustic-level features, e.g., cepstral or LSP coefficients. However, as far as we know, acoustic-level features are the final results of speech production and only represent its surface. There are unobservable features behind them, such as articulatory features. The articulators determine the resonance characteristics of the vocal tract during the production of speech. Therefore, speech can be characterized by both acoustics and vocal-apparatus properties. Articulatory features, which vary much more slowly than acoustic features, seem to be one of the most effective methods of parameterizing speech. Although there have been many attempts at using these parameters in speech synthesis, most of them are rule-based or concatenative (Hill et al., 1995; Sondhi, 2002). If we model articulatory movements by using HMMs and then convert articulatory movements generated from the HMMs with a statistical articulatory-to-acoustic mapping technique (Toda et al., 2004; Nakamura et al., 2006), we can achieve

statistical parametric articulatory synthesis. However, this approach just cascades two independent modules and it may not be an optimal form.

Ling et al. recently proposed an HMM-based acoustic and articulatory joint modeling and synthesis technique to construct statistical parametric articulatory speech synthesis systems (Ling et al., 2008a). The state-output vector of HMMs used in this technique includes both acoustic and articulatory features (static and dynamic). Acoustic and articulatory features were modeled in individual HMM streams and clustered separately by decision trees. Similar to (Hiroya and Honda, 2004), a piece-wise linear transform was adopted to represent the dependence between these two feature streams.[16] This model can be viewed as a special case of factor-analyzed HMMs (FA-HMMs) (Rosti and Gales, 2004). Fig. 15a and b are graphical model representations of HMM and FA-HMM. At the synthesis stage, articulatory and acoustic features are generated simultaneously to maximize their joint-output probability. Synthesized speech in Ling et al.'s technique can be controlled flexibly by modifying the articulatory features generated according to arbitrary phonetic rules during the process of generating parameters. One possible extension of Ling et al.'s technique is using structured speech models, which can include hidden levels of speech production (Richards and Bridle, 1999; Rosti and Gales, 2003; Deng et al., 2006; Frankel and King, 2007; Frankel et al., 2007). Fig. 15c is a graphical model representation of the switching state space model (SSSM), which is a kind of structured speech model. We can see from the figure that the SSSM has additional edges to model the dependencies between $\boldsymbol{g}_{t-1}$ and $\boldsymbol{g}_t$. These models are superb candidates to achieve statistical parametric articulatory speech synthesis. The factor-analyzed trajectory HMM (Toda and Tokuda, 2008) and the joint-probability-modeling technique used in trajectory HMM-based VC (Zen et al., 2008) can also be applied to modeling and synthesizing acoustic and articulatory features.

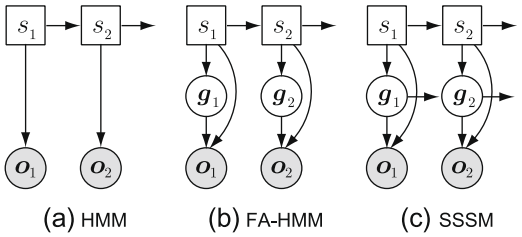

Fig. 15. Graphical model representations of (a) HMM, (b) factor-analyzed (FA) HMM, and (c) switching state space model (SSSM). In the graphical model notation used here, the squares denote discrete variables, the circles continuous variables, the shading an observed variable, and no shading an unobservable variable. The lack of an arc between variables indicates conditional independence. In this figure, s_t is a hidden discrete state, $\boldsymbol{g}_t$ is a hidden continuous vector, and $\boldsymbol{o}_t$ is an observable continuous vector all at time t.

3.3.3. Over-smoothing

In the HMM-based speech synthesis system, the speech parameter generation algorithm (typically the Case 1 algorithm in (Tokuda et al., 2000)) is used to generate spectral and excitation parameters from HMMs to maximize their output probabilities under constraints between static and dynamic features. The statistical averaging in the modeling process improves robustness against data sparseness, and the use of dynamic-feature constraints in the synthesis process enables us to generate smooth trajectories. However, synthesized speech sounds are evidently muffled compared with natural speech because the generated speech-parameter trajectories are often over-smoothed, i.e., detailed characteristics of speech parameters are removed in the modeling part and cannot be recovered in the synthesis part. Although using the advanced acoustic models described in Section 3.3.2 may reduce over-smoothing, this may still exist because the synthesis algorithm does not explicitly include a recovery mechanism.

3.3.3.1. Post-filtering. The simplest way of compensating for over-smoothing is by emphasizing the spectral structure by using a post-filter, which was originally developed for speech coding. The use of post-filtering techniques can reduce "buzziness" and muffled sounds (Yoshimura et al., 2001; Ling et al., 2006; Oura et al., 2007). However, too much post-filtering often introduces artificial sounds and degrades the similarity of synthesized speech to that uttered by the original speaker (Kishimoto et al., 2003).

3.3.3.2. Using real speech data. A second way of compensating for over-smoothing is explicitly using training data to generate parameters. Based on this idea, Masuko et al. proposed a conditional parameter generation algorithm (Masuko et al., 2003). This algorithm generated speech parameters to maximize their output probabilities under additional constraints that some frames in $\boldsymbol{c}$ were fixed as

$$\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{c}} \left\{ \mathcal{N}(\boldsymbol{W}\boldsymbol{c}; \boldsymbol{\mu}_{\boldsymbol{q}}, \boldsymbol{\Sigma}_{\boldsymbol{q}}) \right\}_{\boldsymbol{c}_{t_1}=\tilde{\boldsymbol{c}}_{t_1},\ldots,\boldsymbol{c}_{t_N}=\tilde{\boldsymbol{c}}_{t_N}}, \tag{45}$$

where $\tilde{\boldsymbol{c}}_{t_1}, \ldots, \tilde{\boldsymbol{c}}_{t_N}$ are fixed frames. This is a simple constrained problem, thus we can solve this by using the Lagrange-multiplier method. By copying $\tilde{\boldsymbol{c}}_{t_1}, \ldots, \tilde{\boldsymbol{c}}_{t_N}$ from samples in the training data, we can explicitly use the training data in the generation algorithm. Fig. 16 presents the spectra for natural speech, generated by the conditional speech parameter generation algorithm, and the standard parameter generation algorithm for a sentence included in the training data. We selected the frames around the segment boundaries to be generated in this example. We can see from the figure that we can recover the details of speech spectra in generated frames by fixing other frames with those of natural speech. What is important is how frames to be fixed are selected and how samples in the training

[16] This technique was also applied to model dependence between streams for F_0 and spectral parameters (Ling et al., 2008b).

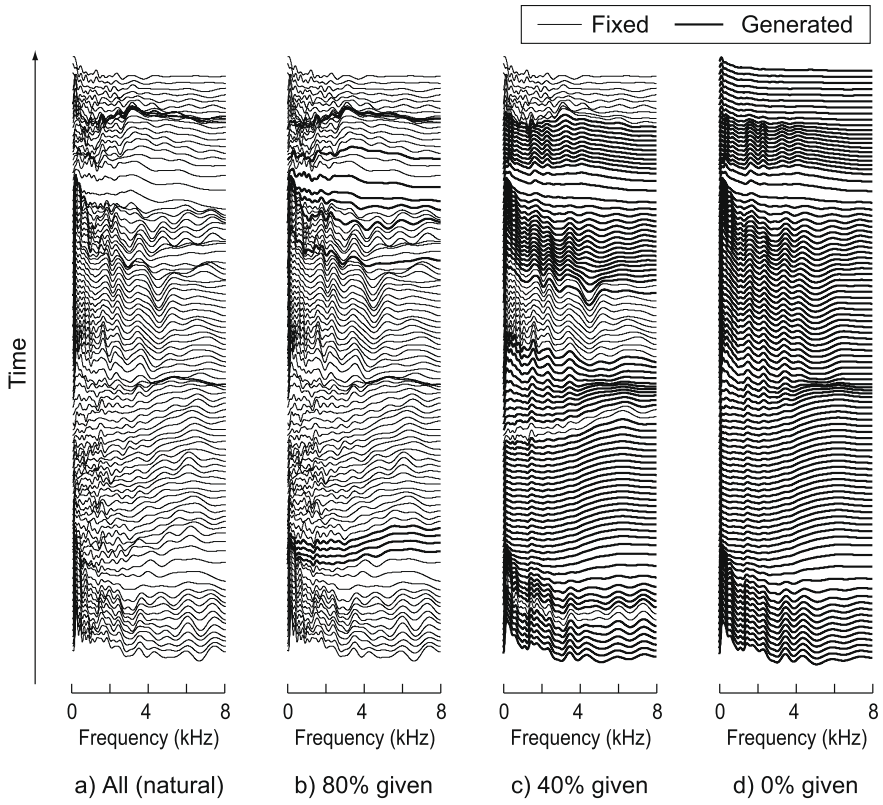

Fig. 16. Spectra generated by conditional parameter generation algorithm. Here (a) all, (b) 80%, (c) 40%, and (d) no frames are given to conditional parameter generation algorithm. Thin lines indicate given frames and thick lines indicate those generated.

data are to be used in this algorithm. Masuko et al. fixed the central frame of each state using a training sample that had the best state-output probability from this state, but their improvements were relatively limited. They reported that frames that did not have spectral details were selected because these samples had better state-output probabilities than those that had spectral details. However, we expect that this algorithm and the smoothing-based hybrid approaches, which will be described in Section 4.2.2, are a good match. Note that discrete HMM-based speech synthesis (Yu et al., 2007) is also based on the same idea (explicitly using training data for generation) to overcome the over-smoothing problem.

3.3.3.3. Using multiple-level statistics. Another way of compensating for over-smoothing is integrating multiple-level statistical models to generate speech-parameter trajectories. Boosting-style additive trees (Qian et al., 2008b), discrete cosine transform (DCT)-based F_0 models (Latorre and Akamine, 2008; Qian et al., 2009), multi-layer F_0 models (Wang et al., 2008), combined multiple-level duration models (Wu and Wang, 2006a; Gao et al., 2008; Qian et al., 2009), and improved intra-phoneme dynamics models (Tiomkin and Malah, 2008) can be categorized as integrated multiple-level statistical models. One of the most successful methods in this category is the speech parameter generation algorithm considering global variance (GV) (Toda and Tokuda, 2007). Fig. 17 shows the trajectories of second mel-cepstral coefficients extracted from natural speech and those generated from an HMM. We can see that the dynamic range of the generated mel-cepstral coefficients is smaller than that of the natural ones. The speech parameter generation algorithm considering GV has focused on solving this phenomenon. It tries to recover the dynamic range of generated trajectories close to those of the natural ones. A GV, $\boldsymbol{v}(\boldsymbol{c})$, is defined as an intra-utterance variance of a speech-parameter trajectory, $\boldsymbol{c}$, as

$$\boldsymbol{v}(\boldsymbol{c}) = [v(1), \ldots, v(M)]^\top, \tag{46}$$

$$v(m) = \frac{1}{T}\sum_{t=1}^{T}\{c_t(m) - \mu(m)\}^2, \tag{47}$$

$$\mu(m) = \frac{1}{T}\sum_{t=1}^{T} c_t(m). \tag{48}$$

We calculate GVs for all training utterances and model them by using a single multi-variate Gaussian distribution as

$$p(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}_{\mathrm{GV}}) = \mathcal{N}(\boldsymbol{v}(\boldsymbol{c}); \boldsymbol{\mu}_{\mathrm{GV}}, \boldsymbol{\Sigma}_{\mathrm{GV}}), \tag{49}$$

where $\boldsymbol{\mu}_{\mathrm{GV}}$ is a mean vector and $\boldsymbol{\Sigma}_{\mathrm{GV}}$ is a covariance matrix of GVs. The speech parameter generation algorithm considering GV maximizes the following objective function with respect to $\boldsymbol{c}$, i.e.,

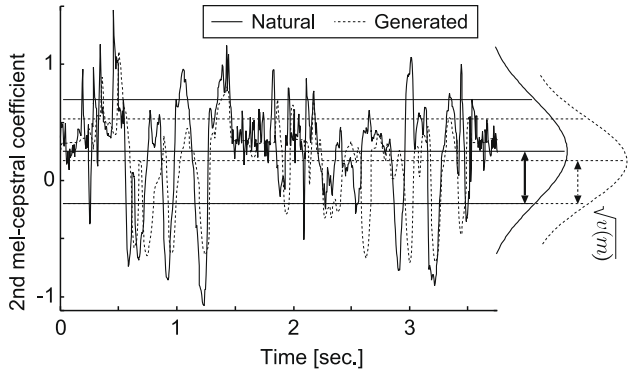

Fig. 17. Trajectories of second mel-cepstral coefficients extracted from natural speech and that generated from HMM. Solid lines indicate natural trajectories and dotted lines indicate those generated.

$$\mathcal{F}_{\mathrm{GV}}(\boldsymbol{c};\boldsymbol{\lambda},\boldsymbol{\lambda}_{\mathrm{GV}}) = \omega \log \mathcal{N}(\boldsymbol{W}\boldsymbol{c};\boldsymbol{\mu}_{\boldsymbol{q}},\boldsymbol{\Sigma}_{\boldsymbol{q}}) + \log \mathcal{N}(\boldsymbol{v}(\boldsymbol{c});\boldsymbol{\mu}_{\mathrm{GV}},\boldsymbol{\Sigma}_{\mathrm{GV}}), \tag{50}$$

where ω is a weight to balance the HMM and GV probabilities. The second term in Eq. (50) can be viewed as a penalty to prevent over-smoothing because it works to retain the dynamic range of the generated trajectory close to that of the training data. This method can be viewed as a statistical post-filtering technique to a certain extent. Fig. 18 has the spectra of natural and synthesized speech generated by the speech parameter generation algorithm, and that considering GV. We can see from the figure that the spectral structure becomes clearer by considering GV. Although it works better than the post-filtering technique (Toda et al., 2007), it still introduces some artificial sounds into synthesized speech (Zen et al., 2006b). To reduce this problem, improved versions of this algorithm have been proposed (Latorre et al., 2007; Yamagishi et al., 2008c). Incorporating GV into the training part of HMM-based speech synthesis has also been proposed (Nakamura, 2007; Wu et al., 2008b; Toda and Young, 2009).

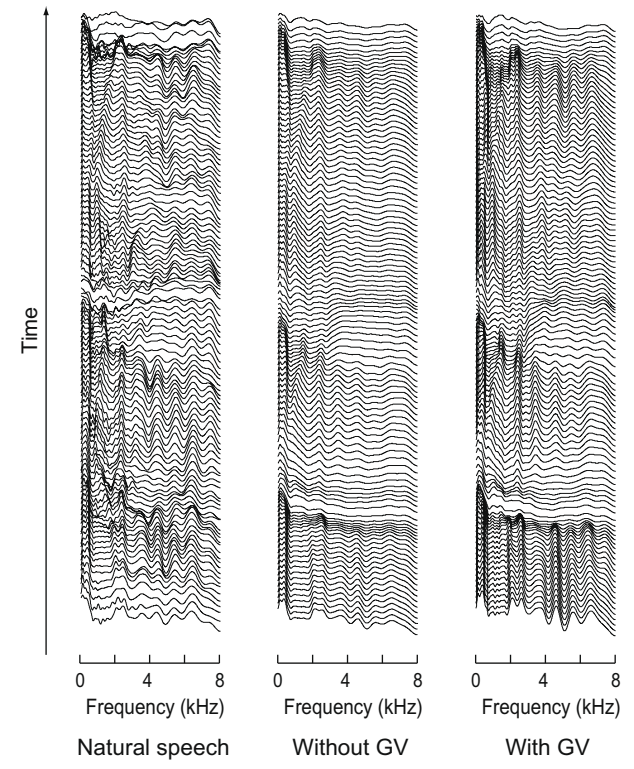

Fig. 18. Spectra of natural and generated speech obtained by speech parameter generation algorithm without and with global variance (GV).

4. Hybrid approaches to statistical parametric and unit-selection synthesis

4.1. Relation between two approaches

Some clustering-based systems for unit selection use HMM-based state clustering (Donovan and Woodland, 1995), where their structure is very similar to that of the HMM-based speech synthesis system. The essential difference between clustering-based unit-selection synthesis and HMM-based speech synthesis is that each cluster in the generation approach is represented by the statistics of the cluster (Fig. 5) instead of the multi-templates of speech units (Fig. 2).

The distributions for the spectrum, excitation (F_0), and duration are clustered independently in the HMM-based speech synthesis system. Therefore, it has different decision trees for each of spectrum, excitation (F_0), and duration (Fig. 19a). However, unit-selection systems often use regression trees (or CART) for predicting prosody. The decision trees for F_0 and duration in the HMM-based speech synthesis system are essentially equivalent to the regression trees in unit-selection systems. However, in the unit-selection systems, the leaves of one of the trees must have speech waveforms; other trees are used to calculate target costs, to prune waveform candidates, or to give features to construct the trees for speech waveforms (Fig. 19b).

It needs to be noted that in HMM-based speech synthesis, the likelihoods of static and dynamic features correspond to the target and concatenation costs. This is easy to understand if we model each state-output distribution with a discrete distribution using vector quantization (VQ) or approximate this by instances of frame samples in the state; when the dynamic feature is calculated as the difference between neighboring static features, ML-based generation results in a frame-wise DP search like the unit selection used in the HMM-based unit-selection system with frame-sized units (Ling and Wang, 2006), i.e.,

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \{C(\boldsymbol{q},\boldsymbol{c})\} \tag{51}$$

$$= \arg\max_{\boldsymbol{c}} \{p(\boldsymbol{c}|\boldsymbol{q},\boldsymbol{\lambda})\}, \tag{52}$$

where

$$C(\boldsymbol{q},\boldsymbol{c}) = \sum_{t=1}^{T} C^{(t)}(q_t,\boldsymbol{c}_t) + \sum_{t=2}^{T} C^{(c)}(\boldsymbol{c}_{t-1},\boldsymbol{c}_t), \tag{53}$$

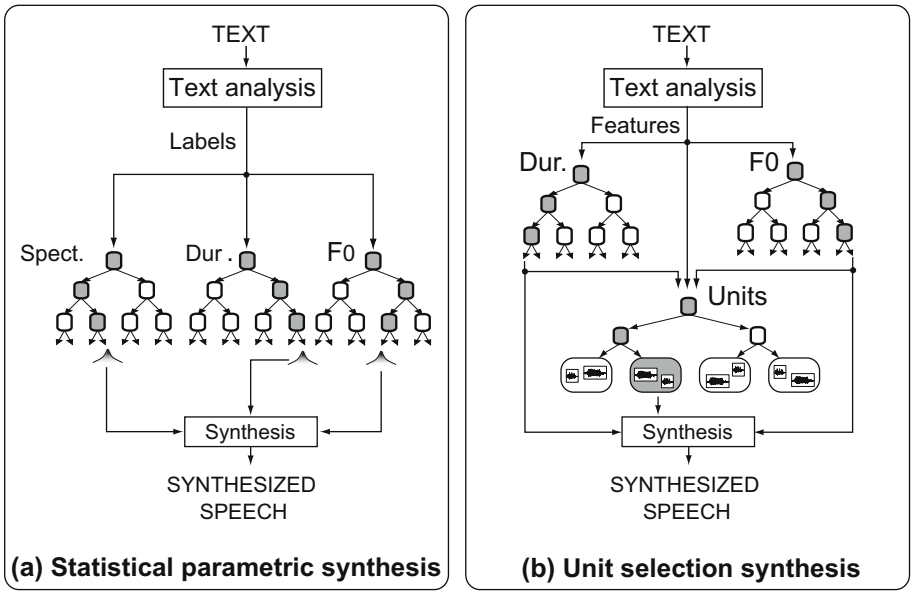

Fig. 19. Overview of use of decision trees in statistical parametric and unit-selection synthesis. Statistical parametric synthesis uses decision trees for spectrum, F_0, and duration in parallel. However, unit-selection synthesis serially cascades F_0, duration, and unit trees.

$$C^{(t)}(q_t, \boldsymbol{c}_t) = -\log \mathcal{N}\left(\boldsymbol{c}_t; \boldsymbol{\mu}^{(s)}_{q_t}, \boldsymbol{\Sigma}^{(s)}_{q_t}\right), \tag{54}$$

$$C^{(c)}(\boldsymbol{c}_{t-1}, \boldsymbol{c}_t) = -\log \mathcal{N}\left(\boldsymbol{c}_t - \boldsymbol{c}_{t-1}; \boldsymbol{\mu}^{(d)}_{q_t}, \boldsymbol{\Sigma}^{(d)}_{q_t}\right), \tag{55}$$

$\boldsymbol{\mu}^{(s)}_j$ and $\boldsymbol{\mu}^{(d)}_j$ correspond to the static- and dynamic-feature parts of $\boldsymbol{\mu}_j$, and $\boldsymbol{\Sigma}^{(s)}_j$ and $\boldsymbol{\Sigma}^{(d)}_j$ correspond to those of $\boldsymbol{\Sigma}_j$. The discrete HMM-based speech synthesis system (Yu et al., 2007) is based on a similar idea. Thus, HMM-based parameter generation can be viewed as an *analogue version* of unit selection.

4.2. Hybrid approaches

There are also hybrid approaches between unit-selection and statistical parametric synthesis as a natural consequence of the viewpoints above.

4.2.1. Target prediction

Some of these approaches use spectrum parameters, F_0 values, and durations (or part of them) generated from HMMs as "targets" for unit-selection synthesis (Kawai et al., 2004; Rouibia and Rosec, 2005; Hirai and Tenpaku, 2004; Yang et al., 2006; Krstulović et al., 2008). Similarly, HMM likelihoods are used as "costs" for unit-selection synthesis (Huang et al., 1996; Hon et al., 1998; Mizutani et al., 2002; Okubo et al., 2006; Ling and Wang, 2006; Ling and Wang, 2007; Ling et al., 2007). Of these approaches, Hirai and Tenpaku (2004) and Ling and Wang (2006) used 5-ms frame-sized units, Huang et al. (1996) and Mizutani et al. (2002) used HMM state-sized units, and Kawai et al. (2004) and Krstulović et al., 2008 used half-phone-sized units. Hon et al. (1998) and Ling et al. (2007) used phone-sized units, Ling and Wang (2007) used hierarchical units consisting of both frame-sized and phone-sized units, Rouibia and Rosec (2005) and Okubo et al. (2006) used diphone-sized units, and Yang et al. (2006) used non-uniform-sized units. Kominek and Black (2006) also used longer trajectories generated from a trajectory model to calculate the costs for unit-selection synthesis.

All these systems used ML-estimated HMMs to predict targets or calculate costs. Ling and Wang recently proposed minimum unit-selection error (MUSE) training (Ling and Wang, 2008) for their HMM-based unit-selection system, which selects a sequence of phone-sized units to maximize the joint-output probability from different sets of HMMs. They defined the unit-selection error as the number of different units between selected and natural unit sequences. Model combination weights and HMM parameters were iteratively optimized to minimize the total unit-selection error by using GPD (Katagiri et al., 1991). They demonstrated that this method could improve the quality of synthesis over the baseline system where model weights are set manually and distribution parameters are trained under the ML criterion. As mentioned in (Ling and Wang, 2008), MUSE training minimizes sentence-level string error. In speech recognition, discriminative training based on fine-grain error measures, such as minimum word error (MWE) or minimum phone error (MPE), often outperforms those based on coarse-grain ones, such as maximum mutual information (MMI) or minimum classification error (MCE) (Povey, 2003). Therefore, we can expect that the use of finer-grain error measures in MUSE training would further improve the quality of synthesis.

Like MUSE training, tightly coupling unit-selection and statistical parametric synthesis techniques are expected to become important to further improve the quality of this type of hybrid systems.

4.2.2. Smoothing units

Another type of hybrid approach uses statistical models and/or dynamic-feature constraints to smooth segment sequences obtained by unit selection.

Plumpe et al. presented a probabilistic framework and the statistics for the smoothing technique for unit-selection synthesis (Plumpe et al., 1998). For a given sentence HMM whose state-output vector includes LSP coefficients and their dynamic features, we can find the trajectories of LSP coefficients that minimize the following objective function:

$$E = \sum_{m=1}^{M} \sum_{t=1}^{T} \frac{\{x_t(m) - \mu_t(m)\}^2}{\sigma_t^2(m)} + D \times \frac{\{x_{t+1}(m) - x_t(m) - \Delta\mu_t(m)\}^2}{\Delta\sigma_t^2(m)}, \tag{56}$$

where D is a constant to control the relative importance of static and dynamic information, and $x_t(m), \mu_t(m)$, and $\sigma_t^2(m)$ correspond to the observation, mean, and variance of the mth LSP coefficient at time t, and $\Delta\mu_t(m)$ and $\Delta\sigma_t^2(m)$ correspond to the mean and variance for the dynamic feature of the mth LSP coefficient at time t. By taking the partial derivative of Eq. (56) with respect to $\{x_t(m)\}$ and equating this to $\mathbf{0}$, we obtain a tri-diagonal set of linear equations to determine $\{x_t(m)\}$, which becomes a special case of Eq. (19) solved in the speech parameter generation algorithm (Tokuda et al., 2000). The above objective function includes the statistics for both static and dynamic features. Therefore, $\{x_t(m)\}$ becomes smooth while retaining static features close to the mean values and maintaining dynamic information. Note that $\{x_t(m)\}$ is identical to the speech-parameter trajectory used in HMM-based speech synthesis if $D = 1$. Instead of using the HMM mean vectors, Plumpe et al. used actual speech segments for $\{\mu_t(m)\}$, to retain the naturalness inherent in unit selection (Plumpe et al., 1998). Smoothing was accomplished by finding $\{x_t(m)\}$ for this $\{\mu_t(m)\}$. Finally, the speech waveform was synthesized from the smoothed LSP coefficients and their residual signals. We can expect dynamic-feature constraints to reduce the discontinuities at the segment boundaries. Plumpe et al. reported that using this smoothing technique reduced spectral discontinuities at segment boundaries where discontinuities should not occur while leaving large spectral jumps where they belonged (Plumpe et al., 1998).

Based on this idea, Wouters and Macon proposed *unit fusion* (Wouters and Macon, 2000). Their system synthesized speech by selecting two types of speech units, i.e., concatenation and fusion. The concatenation units (diphone-sized) specified the initial spectral trajectories, and the fusion units (phoneme-sized) characterized the spectral dynamics at the joining points between concatenation units. After the concatenation and fusion units were selected, they were fused using the information from these with the following objective function

$$\begin{aligned} E = & \sum_{m=1}^{M} \sum_{t=1}^{T} \{x_t(m) - f_t(m)\}^2 \\ & + D_1[\{x_{t+1}(m) - x_t(m)\} - \{f_{t+1}(m) - f_t(m)\}]^2 \\ & + D_2[\{x_t(m+1) - x_t(m)\} - \{f_t(m+1) - f_t(m)\}]^2, \end{aligned} \tag{57}$$

where $\{x_t(m)\}$ are the smoothed LSP coefficients, $\{f_t(m)\}$ are the initial LSP coefficients given by linear interpolation between concatenation and fusion units, and D_1 and D_2 are constants to control relative importance. In Eq. (57), the first term works to retain the smoothed LSP trajectories close to the initial ones, the second obeys dynamic-feature constraints, and the third controls the distance between adjacent LSP coefficients. By taking the partial derivative of Eq. (57) with respect to $\{x_t(m)\}$ and equating it to $\mathbf{0}$, we can obtain a set of linear equations similar to that solved in Plumpe et al.'s smoothing technique. Note that we should minimize this objective function with respect to all LSP coefficients simultaneously because they are dependent on one another through the third term of the objective function. Although this increases the computational cost, it can preserve distances between adjacent LSP coefficients, which is important in human perception. They reported that the unit-fusion approach achieved better objective and subjective scores than time-domain concatenation and linear smoothing.

Although both techniques can reduce the spectral discontinuities at segment boundaries, they introduce some artifacts when there is mismatch between the smoothed filter coefficients and excitation signal.

4.2.3. Mixing natural and generated segments

Yet another hybrid approach is mixing natural and generated segments.

Okubo et al. first proposed this type of hybrid system (Okubo et al., 2006). It first generates a sequence of spectra from a sentence HMM. Then, if there are less than the necessary number of candidate units for the required diphone, a segmental waveform is synthesized for this diphone by arranging the sequence of short-time waveforms obtained by the inverse Fourier transform of generated spectra. Finally, an utterance waveform is synthesized by concatenating the sequences of segmental waveforms obtained from unit selection or parameter generation using pitch-synchronous overlap and add (PSOLA) (Moulines and Charpentier, 1990).

Cereproc's hybrid system proposed by Aylett and Yamagishi is also based on this approach (Aylett and Yamagishi, 2008). If data is sparse and concatenation errors are assessed, it selects a unit sequence from a set of speech segments including synthesized units by statistical parametric synthesis in addition to the standard units. Selected units are then seamlessly concatenated within a unit-selection framework.

Pollet and Breen also proposed this type of hybrid technique that they called multiform segment synthesis, where

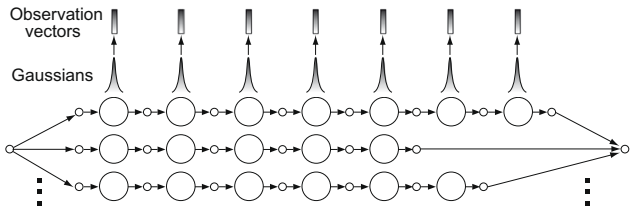

Fig. 20. Huge HMM network representing units. Every unique path represents exactly one unit.

speech parameters are generated and units are selected for a given text to be synthesized (Pollet and Breen, 2008). As a result, two segment sequences are obtained, where the first is by generation and the second is by selection. At the final stage, the best segment sequence is composed to maximize its output probability by selecting either a generated segment or a selected segment while using a speech perception model to assess whether the natural segment is favored or not.

The advantage of this type of hybrid approach is that we can avoid discontinuities due to data sparsity and produce a large proportion of speech while retaining quality of synthesized speech close to that of unit-selection synthesis. However, it also causes quality of synthesized speech to often switch between natural and generated speech. If there is large mismatch between the quality of natural and generated speech segments, frequent switching deteriorates human perception.

4.2.4. Unifying two approaches

Unifying unit-selection and statistical parametric synthesis has also been investigated by Taylor (2006). Let us consider that we have N_u units for a context-dependent sub-word, u, in the training data. Each unit can be represented as a sequence of observation vectors consisting of spectral and excitation parameters. Taylor demonstrated that we can represent these units with a huge HMM network as shown in Fig. 20. In this HMM network, every unique path represents exactly one unit in the training data, and each state-output probability is modeled by a single multi-variate Gaussian distribution whose mean vector is equal to the associated observation vector and the covariance matrix has very small values in its diagonal elements and 0 in its off-diagonal elements. If we synthesize training sentences from this HMM network, we may obtain almost the same speech as from unit selection because this HMM network just memorizes observation vectors by using its topology and statistics. He also explained that we can scale the size of a synthesis system in a principled manner by merging the states of the network, i.e., achieving the same quality as unit selection if no HMM states are merged, and the same quality as a typical HMM-based speech synthesis system if all sub-word HMM networks are merged into the five-state left-to-right HMM structure.

As previously described, there are several types of hybrid approaches between unit-selection and statistical parametric synthesis. In the future, we may converge them into an optimal form of corpus-based speech synthesis fusing statistical parametric and unit-selection synthesis.

5. Conclusion

This review gave a general overview of techniques used in statistical parametric speech synthesis. We can see that statistical parametric synthesis offers a wide range of techniques to improve spoken output. Its more complex models, when compared to unit-selection synthesis, allow for general solutions, without necessarily requiring recorded speech in any phonetic or prosodic contexts. The pure view of unit-selection synthesis requires very large databases to cover examples of all required prosodic, phonetic, and stylistic variations. In contrast, statistical parametric synthesis enables models to be combined and adapted and thus does not require instances of any possible combinations of contexts.

However, there is still much to do in statistical parametric synthesis. As demonstrated in the past Blizzard Challenge events, although the operation of statistical parametric speech synthesis is impressive, its naturalness is still far from that of natural speech (Bennett, 2005; Bennett and Black, 2006; Clark et al., 2007; Karaiskos et al., 2008). Fortunately, as indicated in this review, there are many ideas that have yet to be fully explored and still many more that need to be conceived. When they are, we may find optimal solutions to filling the gap between natural and synthesized speech. There are also numerous possible hybrid approaches between unit-selection and statistical parametric synthesis. As described in this review, unit-selection and statistical parametric synthesis approaches have their own advantages and drawbacks. However, by properly combining these two, we may be able to obtain a first-rate complementary hybrid approach that can solve their respective drawbacks while retaining all their advantages. In the near future, we may find the holy grail of corpus-based speech synthesis fusing statistical parametric and unit-selection synthesis.

Acknowledgements

The authors would like to thank Drs. Tomoki Toda of the Nara Institute of Science and Technology, Junichi Yamagishi of the University of Edinburgh, and Ranniery Maia of the ATR Spoken Language Communication Research Laboratories for their helpful comments and discussions. We are also grateful to many researchers who provided us with useful information that enabled us to write this review. This work was partly supported by the Ministry of Education, Culture, Sports, Science and Technology (MEXT) e-Society project, the Hori information science promotion foundation, a Grant-in-Aid for Scientific Research (No. 1880009) by JSPS, and the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement 213845 (the EMIME Project). This work was also partly supported by the US

National Science Foundation under Grant No. 0415021 "SPICE: Speech Processing Interactive Creation and Evaluation Toolkit for new Languages." Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Abdel-Hamid, O., Abdou, S., Rashwan, M., 2006. Improving Arabic HMM based speech synthesis quality. In: Proc. Interspeech, pp. 1332–1335.

Acero, A., 1999. Formant analysis and synthesis using hidden Markov models. In: Proc. Eurospeech, pp. 1047–1050.

Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Automat. Contr. 19 (6), 716–723.

Akamine, M., Kagoshima, T., 1998. Analytic generation of synthesis units by closed loop training for totally speaker driven text to speech system (TOS drive TTS). In: Proc. ICSLP, pp. 139–142.

Allauzen, C., Mohri, M., Riley, M., 2004. Statistical modeling for unit selection in speech synthesis. In: Proc. 42nd Meeting of the ACL, No. 55.

Anastasakos, T., McDonough, J., Schwartz, R., Makhoul, J., 1996. A compact model for speaker adaptive training. In: Proc. ICSLP, pp. 1137–1140.

Aylett, M., Yamagishi, J., 2008. Combining statistical parametric speech synthesis and unit-selection for automatic voice cloning. In: Proc. LangTech.

Bai, Q., 2007. The development of Chinese TTS technology. Presentation given in SpeechTEK.

Banos, E., Erro, D., Bonafonte, A., Moreno, A., 2008. Flexible harmonic/stochastic modeling for HMM-based speech synthesis. In: V Jornadas en Tecnologías del Habla, pp. 145–148.

Barros, M., Maia, R., Tokuda, K., Freitas, D., Resende Jr., F., 2005. HMM-based European Portuguese speech synthesis. In: Proc. Interspeech, pp. 2581–2584.

Beal, M., 2003. Variational Algorithms for Approximate Bayesian Inference. Ph.D. Thesis, University of London.

Bennett, C., 2005. Large scale evaluation of corpus-based synthesizers: results and lessons from the Blizzard Challenge 2005. In: Proc. Interspeech, pp. 105–108.

Bennett, C., Black, A., 2006. Blizzard Challenge 2006. In: Proc. Blizzard Challenge Workshop.

Berry, J., 2008. Speech synthesis for minority languages: a case study on Scottish Gaelic. In: Proc. Arizona LinguisticsCircle Conference.

Beutnagel, B., Conkie, A., Schroeter, J., Stylianou, Y., Syrdal, A., 1999. The AT&T Next-Gen TTS system. In: Proc. Joint ASA, EAA and DAEA Meeting, pp. 15–19.

Bilmes, J., 2003. Buried Markov models: a graphical modeling approach for automatic speech recognition. Comput. Speech Language 17 (2–3), 213–231.

Black, A., 2002. Perfect synthesis for all of the people all of the time. In: Proc. IEEE Speech Synthesis Workshop.

Black, A., 2003. Unit selection and emotional speech. In: Proc. Eurospeech, pp. 1649–1652.

Black, A., 2006. CLUSTERGEN: a statistical parametric synthesizer using trajectory modeling. In: Proc. Interspeech, pp. 1762–1765.

Black, A., Lenzo, K., 2000. Limited domain synthesis. In: Proc. ICSLP, pp. 411–414.

Black, A., Schultz, T., 2006. Speaker clustering for mulitilingual synthesis. In: Proc. ISCA ITRW MULTILING, No. 024.

Black, A., Taylor, P., 1997. Automatically clustering similar units for unit selection in speech synthesis. In: Proc. Eurospeech, pp. 601–604.

Bonafonte, A., Adell, J., Esquerra, I., Gallego, S., Moreno, A., Pérez, J., 2008. Corpus and voices for Catalan speech synthesis. In: Proc. LREC.

Breen, A., Jackson, P., 1998. A phonologically motivated method of selecting nonuniform units. In: Proc. ICSLP, pp. 2735–2738.

Bulyko, I., Ostendorf, M., Bilmes, J., 2002. Robust splicing costs and efficient search with BMM models for concatenative speech synthesis. In: Proc. ICASSP, pp. 461–464.

Cabral, J., Renals, S., Richmond, K., Yamagishi, J., 2007. Towards an improved modeling of the glottal source in statistical parametric speech synthesis. In: Proc. ISCA SSW6, pp. 113–118.

Cabral, J., Renals, S., Richmond, K., Yamagishi, J., 2008. Glottal spectral separation for parametric speech synthesis. In: Proc. Interspeech, pp. 1829–1832.

Chomphan, S., Kobayashi, T., 2007. Implementation and evaluation of an HMM-based Thai speech synthesis system. In: Proc. Interspeech, pp. 2849–2852.

Clark, R., Podsiadlo, M., Fraser, M., Mayo, C., King, S., 2007. Statistical analysis of the Blizzard Challenge 2007 listening test results. In: Proc. Blizzard Challenge Workshop.

Coorman, G., Fackrell, J., Rutten, P., Coile, B., 2000. Segment selection in the L & H realspeak laboratory TTS system. In: Proc. ICSLP, pp. 395–398.

Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc. 39, 1–38.

Deng, L., 1992. A generalised hidden Markov model with state conditioned trend functions of time for the speech signal. Signal Process. 27 (1), 65–78.

Deng, L., Yu, D., Acero, A., 2006. Structured speech modeling. IEEE Trans. Audio Speech Language Process. 14 (5), 1492–1504.

Dines, J., Sridharan, S., 2001. Trainable speech synthesis with trended hidden Markov models. In: Proc. ICASSP, pp. 833–837.

Donovan, R., Eide, E., 1998. The IBM trainable speech synthesis system. In: Proc. ICSLP, pp. 1703–1706.

Donovan, R., Woodland, P., 1995. Improvements in an HMM-based speech synthesiser. In: Proc. Eurospeech, pp. 573–576.

Drugman, T., Moinet, A., Dutoit, T., 2008. On the use of machine learning in statistical parametric speech synthesis. In: Proc. Benelearn.

Drugman, T., Wilfart, G., Moinet, A., Dutoit, T., 2009. Using a pitch-synchronous residual codebook for hybrid HMM/frame selection speech synthesis. In: Proc. ICASSP, pp. 3793–3796.

Eichner, M., Wolff, M., Hoffmann, R., 2000. A unified approach for speech synthesis and speech recognition using stochastic Markov graphs. In: Proc. ICSLP, pp. 701–704.

Eichner, M., Wolff, M., Ohnewald, S., Hoffman, R., 2001. Speech synthesis using stochastic Markov graphs. In: Proc. ICASSP, pp. 829–832.

Eide, E., Aaron, A., Bakis, R., Hamza, W., Picheny, M., Pitrelli, J., 2004. A corpus-based approach to <AHEM/> expressive speech synthesis. In: Proc. ISCA SSW5.

Fares, T., Khalil, A., Hegazy, A., 2008. Usage of the HMM-based speech synthesis for intelligent Arabic voice. In: Proc. CATA, pp. 93–98.

Ferguson, J., 1980. Variable duration models for speech. In: Proc. Symp. on the Application Hidden Markov Models to Text Speech, pp. 143–179.

Frankel, J., King, S., 2007. Speech recognition using linear dynamic models. IEEE Trans. Speech Audio Process. 15 (1), 246–256.

Frankel, J., Wester, M., King, S., 2007. Articulatory feature recognition using dynamic Bayesian networks. Comput. Speech Language 21 (4), 620–640.

Freij, G., Fallside, F., 1988. Lexical stress recognition using hidden Markov models. In: Proc. ICASSP, pp. 135–138.

Fujinaga, K., Nakai, M., Shimodaira, H., Sagayama, S., 2001. Multiple-regression hidden Markov model. In: Proc. ICASSP, pp. 513–516.

Fukada, T., Tokuda, K., Kobayashi, T., Imai, S., 1992. An adaptive algorithm for mel-cepstral analysis of speech. In: Proc. ICASSP, pp. 137–140.

Gales, M., 1996. The generation and use of regression class trees for MLLR adaptation. Tech. Rep. CUED/F-INFENG/TR263, Cambridge University Engineering Department.

Gales, M., 1998. Maximum likelihood linear transformations for HMM-based speech recognition. Comput. Speech Language 12 (2), 75–98.
Gales, M., 1999. Semi-tied covariance matrices for hidden Markov models. IEEE Trans. Speech Audio Process. 7 (3), 272–281.
Gales, M., 2000. Cluster adaptive training of hidden Markov models. IEEE Trans. Speech Audio Process. 8 (4), 417–428.
Gao, B.-H., Qian, Y., Wu, Z.-Z., Soong, F.-K., 2008. Duration refinement by jointly optimizing state and longer unit likelihood. In: Proc. Interspeech, pp. 2266–2269.
Gauvain, J., Lee, C.-H., 1994. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. IEEE Trans. Speech Audio Process. 2 (2), 291–298.
Gish, H., Ng, K., 1993. A segmental speech model with application to word spotting. In: Proc. ICASSP, pp. 447–450.
Gonzalvo, X., Iriondo, I., Socoró, Alías, F., Monzo, C., 2007a. HMM-based Spanish speech synthesis using CBR as F0 estimator. In: Proc. NOLISP, pp. 7–10.
Gonzalvo, X., Socoró, C., Iriondo, I., Monzo, C., Martínez, E., 2007b. Linguistic and mixed excitation improvements on a HMM-based speech synthesis for Castilian Spanish. In: Proc. ISCA SSW6, pp. 362–367.
Hashimoto, K., Zen, H., Nankaku, Y., Tokuda, K., 2008. HMM-based speech synthesis using cross validation for Bayesian criterion. In: Proc. Autumn Meeting of ASJ, pp. 251–252 (in Japanese).
Hemptinne, C., 2006. Integration of the Harmonic Plus Noise Model into the Hidden Markov Model-Based Speech Synthesis System. Master Thesis, IDIAP Research Institute.
Hill, D., Manzara, L., Schock, C., 1995. Real-time articulatory speech-synthesis-by-rules. In: Proc. AVIOS Symposium, pp. 27–44.
Hirai, T., Tenpaku, S., 2004. Using 5 ms segments in concatenative speech synthesis. In: Proc. ISCA SSW5.
Hirose, K., Sato, K., Asano, Y., Minematsu, N., 2005. Synthesis of f_0 contours using generation process model parameters predicted from unlabeled corpora: application to emotional speech synthesis. Speech Comm. 46 (3–4), 385–404.
Hiroya, S., Honda, M., 2004. Estimation of articulatory movements from speech acoustics using an HMM-based speech production model. IEEE Trans. Speech Audio Process. 12 (2), 175–185.
Homayounpour, M., Mehdi, S., 2004. Farsi speech synthesis using hidden Markov model and decision trees. CSI J. Comput. Sci. Eng. 2 (1&3 (a)) (in Farsi).
Hon, H.-W., Acero, A., Huang, X.-D., Liu, J.-S., Plumpe, M., 1998. Automatic generation of synthesis units for trainable text-to-speech systems. In: Proc. ICASSP, pp. 293–296.
Huang, X.-D., Acero, A., Adcock, J., Hon, H.-W., Goldsmith, J., Liu, J.-S., 1996. Whistler: a trainable text-to-speech system. In: Proc. ICSLP, pp. 2387–2390.
Hunt, A., Black, A., 1996. Unit selection in a concatenative speech synthesis system using a large speech database. In: Proc. ICASSP, pp. 373–376.
Imai, S., Sumita, K., Furuichi, C., 1983. Mel log spectrum approximation (MLSA) filter for speech synthesis. Electron. Comm. Jpn. 66 (2), 10–18.
Irino, T., Minami, Y., Nakatani, T., Tsuzaki, M., Tagawa, H., 2002. Evaluation of a speech recognition/ generation method based on HMM and STRAIGHT. In: Proc. ICSLP, pp. 2545–2548.
Ishimatsu, Y., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 2001. Investigation of state duration model based on gamma distribution for HMM-based speech synthesis. In: Tech. Rep. of IEICE, SP2001-81, Vol. 101, pp. 57–62 (in Japanese).
Itakura, F., 1975. Minimum prediction residual principle applied to speech recognition. IEEE Trans. Acoust. Speech Signal Process. 23 (1), 67–72.
Iwahashi, N., Sagisaka, Y., 1995. Speech spectrum conversion based on speaker interpolation and multi-functional representation with weighting by radial basis function networks. Speech Comm. 16 (2), 139–151.
Iwano, K., Yamada, M., Togawa, T., Furui, S., 2002. Speech rate control for HMM-based speech synthesis. In: Tech. Rep. of IEICE, No. SP2002-73, pp. 11–16.
Jensen, U., Moore, R., Dalsgaard, P., Lindberg, B., 1994. Modeling intonation contours at the phrase level using continuous density hidden Markov models. Comput. Speech Language 8 (3), 247–260.
Juang, B.-H., Chou, W., Lee, C.-H., 1997. Minimum classification error rate methods for speech recognition. IEEE Trans. Speech Audio Process. 5 (3), 257–265.
Karabetsos, S., Tsiakoulis, P., Chalamandaris, A., Raptis, S., 2008. HMM-based speech synthesis for the Greek language. In: Proc. TSD, pp. 349–356.
Karaiskos, V., King, S., Clark, R., Mayo, C., 2008. The Blizzard Challenge 2008. In: Proc. Blizzard Challenge Workshop.
Katagiri, S., Lee, C.-H., Juang, B.-H., 1991. New discriminative training algorithms based on the generalized probabilistic descent method. In: Proc. IEEE Internat. Workshop Neural Networks for Signal Process, pp. 299–308.
Kataoka, S., Mizutani, N., Tokuda, K., Kitamura, T., 2004. Decision-tree backing-off in HMM-based speech synthesis. In: Proc. Interspeech, pp. 1205–1208.
Kawahara, H., Masuda-Katsuse, I., Cheveigne, A., 1999. Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based f0 extraction: possible role of a repetitive structure in sounds. Speech Comm. 27 (3), 187–207.
Kawai, H., Tsuzaki, M., 2002. A study on time-dependent voice quality variation in a large-scale single speaker speech corpus used for speech synthesis. In: Proc. IEEE Speech Synthesis Workshop.
Kawai, H., Toda, T., Ni, J., Tsuzaki, M., Tokuda, K., 2004. XIMERA: a new TTS from ATR based on corpus-based technologies. In: Proc. ISCA SSW5.
KDDI R&D Laboratories, 2008. Development of downloadable speech synthesis software for mobile phones. Press release. <http://www.kddilabs.jp/press/detail_100.html>.
Kim, S.-J., Hahn, M.-S., 2007. Two-band excitation for HMM-based speech synthesis. IEICE Trans. Inform. Systems E90-D (1), 378–381.
Kim, S.-J., Kim, J.-J., Hahn, M.-S., 2006a. HMM-based Korean speech synthesis system for hand-held devices. IEEE Trans. Consumer Electron. 52 (4), 1384–1390.
Kim, S.-J., Kim, J.-J., Hahn, M.-S., 2006b. Implementation and evaluation of an HMM-based Korean speech synthesis system. IEICE Trans. Inform. Systems, 1116–1119.
King, S., Tokuda, K., Zen, H., Yamagishi, J., 2008. Unsupervised adaptation for HMM-based speech synthesis. In: Proc. Interspeech, pp. 1869–1872.
Kishimoto, Y., Zen, H., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 2003. Automatic estimation of postfilter coefficients for HMM-based speech synthesis. In: Proc. Spring Meeting of ASJ, pp. 243–244 (in Japanese).
Kishore, S., Black, A., 2003. Unit size in unit selection speech synthesis. In: Proc. Interspeech, pp. 1317–1320.
Koishida, K., Tokuda, K., Masuko, T., Kobayashi, T., 2001. Vector quantization of speech spectral parameters using statistics of static and dynamic features. IEICE Trans. Inform. Systems E84-D (10), 1427–1434.
Kominek, J., Black, A., 2003. CMU ARCTIC databases for speech synthesis. Tech. Rep. CMU-LTI-03-177, Carnegie Mellon University.
Kominek, J., Black, A., 2006. The Blizzard Challenge 2006 CMU entry introducing hybrid trajectory-selection synthesis. In: Proc. Blizzard Challenge Workshop.
Krstulović, S., Hunecke, A., Schröder, M., 2007. An HMM-based speech synthesis system applied to German and its adaptation to a limited set of expressive football announcements. In: Proc. Interspeech, pp. 1897–1900.
Krstulović, S., Latorre, J., Buchholz, S., 2008. Comparing QMT1 and HMMs for the synthesis of American English prosody. In: Proc. Speech Prosody, pp. 67–70.
Kuhn, R., Janqua, J., Nguyen, P., Niedzielski, N., 2000. Rapid speaker adaptation in eigenvoice space. IEEE Trans. Speech Audio Process. 8 (6), 695–707.
Latorre, J., Akamine, M., 2008. Multilevel parametric-base F0 model for speech synthesis. In: Proc. Interspeech, pp. 2274–2277.

Latorre, J., Iwano, K., Furui, S., 2006. New approach to the polyglot speech generation by means of an HMM-based speaker adaptable synthesizer. Speech Comm. 48 (10), 1227–1242.

Latorre, J., Iwano, K., Furui, S., 2007. Combining Gaussian mixture model with global variance term to improve the quality of an HMM-based polyglot speech synthesizer. In: Proc. ICASSP, pp. 1241–1244.

Leggetter, C., Woodland, P., 1995. Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. Comput. Speech Language 9, 171–185.

Levinson, S., 1986. Continuously variable duration hidden Markov models for automatic speech recognition. Comput. Speech Language 1, 29–45.

Liang, H., Qian, Y., Soong, F.-K., Liu, G., 2008. A cross-language state mapping approach to bilingual (Mandarin–English) TTS. In: Proc. ICASSP, pp. 4641–4644.

Ling, Z.-H., Wang, R.-H., 2006. HMM-based unit selection using frame sized speech segments. In: Proc. Interspeech, pp. 2034–2037.

Ling, Z.-H., Wang, R.-H., 2007. HMM-based hierarchical unit selection combining Kullback–Leibler divergence with likelihood criterion. In: Proc. ICASSP, pp. 1245–1248.

Ling, Z.-H., Wang, R.-H., 2008. Minimum unit selection error training for HMM-based unit selection speech synthesis system. In: Proc. ICASSP, pp. 3949–3952.

Ling, Z.-H., Wu, Y.-J., Wang, Y.-P., Qin, L., Wang, R.-H., 2006. USTC system for Blizzard Challenge 2006 – an improved HMM-based speech synthesis method. In: Proc. Blizzard Challenge Workshop.

Ling, Z.-H., Qin, L., Lu, H., Gao, Y., Dai, L.-R., Wang, R.-H., Jian, Y., Zhao, Z.-W., Yang, J.-H., Chen, J., Hu, G.-P., 2007. The USTC and iFlytek speech synthesis systems for Blizzard Challenge 2007. In: Proc. Blizzard Challenge Workshop.

Ling, Z.-H., Richmond, K., Yamagishi, J., Wang, R.-H., 2008a. Articulatory control of HMM-based parametric speech synthesis driven by phonetic knowledge. In: Proc. Interspeech, pp. 573–576.

Ling, Z.-H., Zhang, W., Wang, R.-H., 2008b. Cross-stream dependency modeling for HMM-based speech synthesis. In: Proc. ISCSLP, pp. 5–8.

Lu, H., Wu, Y.-J., Tokuda, K., Dai, L.-R., Wang, R.-H., 2009. Full covariance state duration modeling for HMM-based speech synthesis. In: Proc. ICASSP, pp. 4033–4036.

Lundgren, A., 2005. An HMM-Based Text-to-Speech System Applied to Swedish. Master Thesis, Royal Institute of Technology (KTH) (in Swedish).

Maia, R., Zen, H., Tokuda, K., Kitamura, T., Resende Jr., F., 2003. Towards the development of a Brazilian Portuguese text-to-speech system based on HMM. In: Proc. Eurospeech, pp. 2465–2468.

Maia, R., Toda, T., Zen, H., Nankaku, Y., Tokuda, K., 2007. An excitation model for HMM-based speech synthesis based on residual modeling. In: Proc. ISCA SSW6, pp. 131–136.

Maia, R., Toda, T., Tokuda, K., Sakai, S., Nakamura, S., 2008. On the state definition for a trainable excitation model in HMM-based speech synthesis. In: Proc. ICASSP, pp. 3965–3968.

Maia, R., Toda, T., Tokuda, K., Sakai, S., Shimizu, T., Nakamura, S., 2009. A decision tree-based clustering approach to state definition in a residual modeling framework. In: Proc. Spring Meeting of ASJ, pp. 311–312.

Martincic-Ipsic, S., Ipsic, I., 2006. Croatian HMM-based speech synthesis. J. Comput. Inform. Technol. 14 (4), 307–313.

Marume, M., Zen, H., Nankaku, Y., Tokuda, K., Kitamura, T., 2006. An investigation of spectral parameters for HMM-based speech synthesis. In: Proc. Autumn Meeting of ASJ, pp. 185–186 (in Japanese).

Masuko, T., Tokuda, K., Kobayashi, T., Imai, S., 1997. Voice characteristics conversion for HMM-based speech synthesis system. In: Proc. ICASSP, pp. 1611–1614.

Masuko, T., Tokuda, K., Kobayashi, T., 2003. A study on conditional parameter generation from HMM based on maximum likelihood criterion. In: Proc. Autumn Meeting of ASJ, pp. 209–210 (in Japanese).

Matsuda, S., Nakai, M., Shimodaira, H., Sagayama, S., 2003. Speech recognition using asynchronous transition HMM. IEICE Trans. Inform. Systems J86-DII (6), 741–754.

Miyanaga, K., Masuko, T., Kobayashi, T., 2004. A style control technique for HMM-based speech synthesis. In: Proc. Interspeech, pp. 1437–1439.

Mizutani, N., Tokuda, K., Kitamura, T., 2002. Concatenative speech synthesis based on HMM. In: Proc. Autumn Meeting of ASJ, pp. 241–242 (in Japanese).

Morioka, Y., Kataoka, S., Zen, H., Nankaku, Y., Tokuda, K., Kitamura, T., 2004. Miniaturization of HMM-based speech synthesis. In: Proc. Autumn Meeting of ASJ, pp. 325–326 (in Japanese).

Moulines, E., Charpentier, F., 1990. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. Speech Comm. 9, 453–467.

Nakamura, K., 2007. Acoustic Model Training Considering Global Variance for HMM-Based Speech Synthesis. Master Thesis, Nagoya Institute of Technology (in Japanese).

Nakamura, K., Toda, T., Nankaku, Y.K.T., 2006. On the use of phonetic information for mapping from articulatory movements to vocal tract spectrum. In: Proc. ICASSP, pp. 93–96.

Nakatani, N., Yamamoto, K., Matsumoto, H., 2006. Mel-LSP parameterization for HMM-based speech synthesis. In: Proc. SPECOM, pp. 261–264.

Nankaku, Y., Zen, H., Tokuda, K., Kitamura, T., Masuko, T., 2003. A Bayesian approach to HMM-based speech synthesis. In: Tech. Rep. of IEICE, Vol. 103, pp. 19–24 (in Japanese).

Nose, T., Kato, Y., Kobayashi, T., 2007a. Style estimation of speech based on multiple regression hidden semi-Markov model. In: Proc. ISCA SSW6, pp. 2285–2288.

Nose, T., Yamagishi, J., Masuko, T., Kobayashi, T., 2007b. A style control technique for HMM-based expressive speech synthesis. IEICE Trans. Inform. Systems E90-D (9), 1406–1413.

Nose, T., Tachibana, M., Kobayashi, T., 2009. HMM-based style control for expressive speech synthesis with arbitrary speaker's voice using model adaptation. IEICE Trans. Inform. Systems E92-D (3), 489–497.

Odell, J., 1995. The Use of Context in Large Vocabulary Speech Recognition. Ph.D. Thesis, University of Cambridge.

Ogata, K., Tachibana, M., Yamagishi, J., Kobayashi, T., 2006. Acoustic model training based on linear transformation and MAP modification for HSMM-based speech synthesis. In: Proc. Interspeech, pp. 1328–1331.

Ojala, T., 2006. Auditory Quality Evaluation of Present Finnish Text-to-Speech Systems. Master Thesis, Helsinki University of Technology.

Okubo, T., Mochizuki, R., Kobayashi, T., 2006. Hybrid voice conversion of unit selection and generation using prosody dependent HMM. IEICE Trans. Inform. Systems E89-D (11), 2775–2782.

Olsen, P., Gopinath, R., 2004. Modeling inverse covariance matrices by basis expansion. IEEE Trans. Acoust. Speech Signal Process. 12, 37–46.

Oura, K., Zen, H., Nankaku, Y., Lee, A., Tokuda, K., 2007. Postfiltering for HMM-based speech synthesis using mel-LSPs. In: Proc. Autumn Meeting of ASJ, pp. 367–368 (in Japanese).

Oura, K., Nankaku, Y., Toda, T., Tokuda, K., Maia, R., Sakai, S., Nakamura, S., 2008a. Simultaneous acoustic, prosodic, and phrasing model training for TTS conversion systems. In: Proc. ISCSLP, pp. 1–4.

Oura, K., Zen, H., Nankaku, Y., Lee, A., Tokuda, K., 2008b. Tying variance for HMM-based speech synthesis. In: Proc. Autumn Meeting of ASJ, pp. 421–422 (in Japanese).

Penny, W., Roberts, S., 1998. Hidden Markov models with extended observation densities. Tech. Rep., Neural Systems Research Group, Imperial College of Science, Technology and Medicine.

Plumpe, M., Acero, A., Hon, H.-W., Huang, X.-D., 1998. HMM-based smoothing for concatenative speech synthesis. In: Proc. ICSLP, pp. 2751–2754.

Pollet, V., Breen, A., 2008. Synthesis by generation and concatenation of multiform segments. In: Proc. Interspeech, pp. 1825–1828.

Povey, D., 2003. Discriminative Training for Large Vocabulary Speech Recognition. Ph.D. Thesis, University of Cambridge.

Qian, Y., Soong, F.-K., Chen, Y., Chu, M., 2006. An HMM-based Mandarin Chinese text-to-speech system. In: Proc. ISCSLP, pp. 223–232.

Qian, Y., Cao, H.-W., Soong, F.-K., 2008a. HMM-based mixed-language (Mandarin–English) speech synthesis. In: Proc. ISCSLP, pp. 13–16.
Qian, Y., Liang, H., Soong, F.-K., 2008b. Generating natural F0 trajectory with additive trees. In: Proc. Interspeech, pp. 2126–2129.
Qian, Y., Wu, Z.-Z., Soong, F.-K., 2009. Improved prosody generation by maximizing joint likelihood of state and longer units. In: Proc. ICASSP, pp. 3781–3784.
Qin, L., Wu, Y.-J., Ling, Z.-H., Wang, R.-H., 2006. Improving the performance of HMM-based voice conversion using context clustering decision tree and appropriate regression matrix format. In: Proc. Interspeech, pp. 2250–2253.
Qin, L., Wu, Y.-J., Ling, Z.-H., Wang, R.-H., Dai, L.-R., 2008. Minimum generation error linear regression based model adaptation for HMM-based speech synthesis. In: Proc. ICASSP, pp. 3953–3956.
Raitio, T., Suni, A., Pulakka, H., Vainio, M., Alku, P., 2008. HMM-based Finnish text-to-speech system utilizing glottal inverse filtering. In: Proc. Interspeech, pp. 1881–1884.
Richards, H., Bridle, J., 1999. The HDM: a segmental hidden dynamic model of coarticulation. In: Proc. ICASSP, Vol. 1, pp. 357–360.
Rissanen, J., 1980. Stochastic Complexity in Stochastic Inquiry. World Scientific Publishing Company.
Ross, K., Ostendorf, M., 1994. A dynamical system model for generating F0 for synthesis. In: Proc. ESCA/IEEE Workshop on Speech Synthesis, pp. 131–134.
Rosti, A., Gales, M., 2003. Switching linear dynamical systems for speech recognition. Tech. Rep. CUED/F-INFENG/TR.461, University of Cambridge.
Rosti, A.-V., Gales, M., 2004. Factor analysed hidden Markov models for speech recognition. Comput. Speech Language 18 (2), 181–200.
Rouibia, S., Rosec, O., 2005. Unit selection for speech synthesis based on a new acoustic target cost. In: Proc. Interspeech, pp. 2565–2568.
Russell, M., Moore, R., 1985. Explicit modeling of state occupancy in hidden Markov models for automatic speech recognition. In: Proc. ICASSP, pp. 5–8.
Sagisaka, Y., Kaiki, N., Iwahashi, N., Mimura, K., 1992. ATR ν-TALK speech synthesis system. In: Proc. ICSLP, pp. 483–486.
Sakai, S., Shu, H., 2005. A probabilistic approach to unit selection for corpus-based speech synthesis. In: Proc. Interspeech, pp. 81–84.
Sakti, S., Maia, R., Sakai, S., Shimizu, T., Nakamura, S., 2008. Development of HMM-based Indonesian speech synthesis. In: Proc. Oriental COCOSDA, pp. 215–219.
Schwarz, G., 1978. Estimating the dimension of a model. Ann. Statist. 6 (2), 461–464.
Segi, H., Takagi, T., Ito, T., 2004. A concatenative speech synthesis method using context dependent phoneme sequences with variable length as search units. In: Proc. ISCA SSW5, pp. 115–120.
Sherpa, U., Pemo, D., Chhoeden, D., Rugchatjaroen, A., Thangthai, A., Wutiwiwatchai, C., 2008. Pioneering Dzongkha text-to-speech synthesis. In: Proc. Oriental COCOSDA, pp. 150–154.
Shichiri, K., Sawabe, A., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 2002. Eigenvoices for HMM-based speech synthesis. In: Proc. ICSLP, pp. 1269–1272.
Shi, Y., Chang, E., Peng, H., Chu, M., 2002. Power spectral density based channel equalization of large speech database for concatenative TTS system. In: Proc. ICSLP, pp. 2369–2372.
Shinoda, K., Lee, C.-H., 2001. A structural Bayes approach to speaker adaptation. IEEE Trans. Speech Audio Process. 9, 276–287.
Shinoda, K., Watanabe, T., 2000. MDL-based context-dependent subword modeling for speech recognition. J. Acoust. Soc. Jpn. (E) 21 (2), 79–86.
Shinta, Y., Nakai, M., Shimodaira, H., 2005. Asynchronous transition HMM-based speech synthesis. In: Proc. Research Workshop of Hokuriku-Area Students No. F-25.
Silen, H., Helander, E., Nurminen, J., Gabbouj, M., 2008. Evaluation of Finnish unit selection and HMM-based speech synthesis. In: Proc. Interspeech, pp. 1853–1856.
Sondhi, M., 2002. Articulatory modeling: a possible role in concatenative text-to-speech synthesis. In: Proc. IEEE Speech Synthesis Workshop.
Stylianou, Y., 1999. Assessment and correction of voice quality variabilities in large speech databases for concatenative speech synthesis. In: Proc. ICASSP, pp. 377–380.
Stylianou, Y., Capṕe, O., Moulines, E., 1998. Continuous probabilistic transform for voice conversion. IEEE Trans. Speech Audio Process. 6 (2), 131–142.
Sun, J.-W., Ding, F., Wu, Y.-H., 2009. Polynomial segment model based statistical parametric speech synthesis system. In: Proc. ICASSP, pp. 4021–4024.
SVOX AG, 2007. SVOX announces SVOX Pico, a revolutionary new hidden Markov model-based text-to-speech product for mobile phones. Press Release. <http://www.svox.com/upload/pdf/PR_SVOX_Pico.pdf>.
SVOX AG, 2008. SVOX releases Pico: highest-quality sub-1MB TTS. Press Release. <http://www.svox.com/upload/pdf/PR_SVOX_Pico_Release_Nov_08.pdf>.
Sýkora, T., 2006. Syntéza slovenskej reči. Master Thesis, Slovak University of Technology (in Slovak).
Tachibana, M., Yamagishi, J., Masuko, T., Kobayashi, T., 2005. Speech synthesis with various emotional expressions and speaking styles by style interpolation and morphing. IEICE Trans. Inform. Systems E88-D (11), 2484–2491.
Tachibana, M., Yamagishi, J., Masuko, T., Kobayashi, T., 2006. A style adaptation technique for speech synthesis using HSMM and suprasegmental features. IEICE Trans. Inform. Systems E89-D (3), 1092–1099.
Tachibana, M., Izawa, S., Nose, T., Kobayashi, T., 2008. Speaker and style adaptation using average voice model for style control in HMM-based speech synthesis. In: Proc. ICASSP, pp. 4633–4636.
Tachiwa, W., Furui, S., 1999. A study of speech synthesis using HMMs. In: Proc. Spring Meeting of ASJ, pp. 239–240 (in Japanese).
Takahashi, J., Sagayama, S., 1995. Vector-field-smoothed Bayesian learning for incremental speaker adaptation. In: Proc. ICASSP, pp. 696–699.
Takahashi, T., Tokuda, K., Kobayashi, T., Kitamura, T., 2001. Training algorithm of HMMs based on mel-cepstral representation. In: Proc. Autumn Meeting of ASJ, Vol. 1, pp. 5–6 (in Japanese).
Tamura, M., Masuko, T., Tokuda, K., Kobayashi, T., 2001. Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR. In: Proc. ICASSP, pp. 805–808.
Tamura, M., Mizutani, T., Kagoshima, T., 2005. Scalable concatenative speech synthesis based on the plural unit selection and fusion method. In: Proc. ICASSP, pp. 351–354.
Taylor, P., 2006. Unifying unit selection and hidden Markov model speech synthesis. In: Proc. Interspeech, pp. 1758–1761.
Taylor, P., Black, A., 1999. Speech synthesis by phonological structure matching. In: Proc. Eurospeech, pp. 1531–1534.
Tiomkin, S., Malah, D., 2008. Statistical text-to-speech synthesis with improved dynamics. In: Proc. Interspeech, pp. 1841–1844.
Toda, T., Tokuda, K., 2007. A speech parameter generation algorithm considering global variance for HMM-based speech synthesis. IEICE Trans. Inform. Systems E90-D (5), 816–824.
Toda, T., Tokuda, K., 2008. Statistical approach to vocal tract transfer function estimation based on factor analyzed trajectory HMM. In: Proc. ICASSP, pp. 3925–3928.
Toda, T., Young, S., 2009. Trajectory training considering global variance for HMM-based speech synthesis. In: Proc. ICASSP, pp. 4025–4028.
Toda, T., Black, A., Tokuda, K., 2004. Mapping from articulatory movements to vocal tract spectrum with Gaussian mixture model for articulatory speech synthesis. In: Proc. ISCA SSW5, pp. 31–36.
Toda, T., Black, A., Tokuda, K., 2007. Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory. IEEE Trans. Audio Speech Language Process. 15 (8), 2222–2235.
Tokuda, K., Black, A., 2005. The Blizzard Challenge 2005: evaluating corpus-based speech synthesis on common datasets. In: Proc. Interspeech, pp. 77–80.
Tokuda, K., Kobayashi, T., Imai, S., 1995. Speech parameter generation from hmm using dynamic features. In: Proc. ICASSP, pp. 660–663.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., Kitamura, T., 2000. Speech parameter generation algorithms for HMM-based speech synthesis. In: Proc. ICASSP, pp. 1315–1318.

Tokuda, K., Masuko, T., Miyazaki, N., Kobayashi, T., 2002a. Multi-space probability distribution HMM. IEICE Trans. Inform. Systems E85-D (3), 455–464.

Tokuda, K., Zen, H., Black, A., 2002b. An HMM-based speech synthesis system applied to English. In: Proc. IEEE Speech Synthesis Workshop.

Tokuda, K., Zen, H., Yamagishi, J., Black, A., Masuko, T., Sako, S., Toda, T., Nose, T., Oura, K., 2008. The HMM-based speech synthesis system (HTS). <http://hts.sp.nitech.ac.jp/>.

Tóth, B., Németh, G., 2008. Hidden Markov model based speech synthesis system in Hungarian. Infocommunications 63 (7), 30–34.

Vainio, M., Suni, A., Sirjola, P., 2005. Developing a Finnish concept-to-speech system. In: Proc. 2nd Baltic Conf. on HLT, pp. 201–206.

Vesnicer, B., Mihelic, F., 2004. Evaluation of the Slovenian HMM-based speech synthesis system. In: Proc. TSD, pp. 513–520.

Wang, C.-C., Ling, Z.-H., Zhang, B.-F., Dai, L.-R., 2008. Multi-layer F0 modeling for HMM-based speech synthesis. In: Proc. ISCSLP, pp. 129–132.

Watanabe, S., 2007. Almost all learning machines are singular. In: Proc. IEEE Symp. on Foundations of Computational Intelligence, pp. 383–388.

Watanabe, S., Minami, Y., Nakamura, A., Ueda, N., 2004. Variational Bayesian estimation and clustering for speech recognition. IEEE Trans. Speech Audio Process. 12 (4), 365–381.

Watanabe, T., Zen, H., Nankaku, Y., Lee, A., Tokuda, K., 2007. Reducing computational complexity of a synthesis filter for HMM-based speech synthesis. In: Proc. Autumn Meeting of ASJ, pp. 209–210 (in Japanese).

Weiss, C., Maia, R., Tokuda, K., Hess, W., 2005. Low resource HMM-based speech synthesis applied to German. In: Proc. ESSP.

Wouters, J., Macon, M., 2000. Unit fusion for concatenative speech synthesis. In: Proc. ICSLP, pp. 302–305.

Wu, Y.-J., Tokuda, K., 2008. An improved minimum generation error training with log spectral distortion for HMM-based speech synthesis. In: Proc. Interspeech, pp. 577–580.

Wu, Y.-J., Tokuda, K., 2009. Minimum generation error training by using original spectrum as reference for log spectral distortion measure. In: Proc. ICASSP, pp. 4013–4016.

Wu, Y.-J., Wang, R.-H., 2006a. HMM-based trainable speech synthesis for Chinese. J. Chin. Inform. Process. 20 (4), 75–81 (in Chinese).

Wu, Y.-J., Wang, R.-H., 2006b. Minimum generation error training for HMM-based speech synthesis. In: Proc. ICASSP, pp. 89–92.

Wu, Y.-J., Guo, W., Wang, R.-H., 2006. Minimum generation error criterion for tree-based clustering of context dependent HMMs. In: Proc. Interspeech, pp. 2046–2049.

Wu, Y.-J., King, S., Tokuda, K., 2008a. Cross-language speaker adaptation for HMM-based speech synthesis. In: Proc. ISCSLP, pp. 9–12.

Wu, Y.-J., Zen, H., Nankaku, Y., Tokuda, K., 2008b. Minimum generation error criterion considering global/local variance for HMM-based speech synthesis. In: Proc. ICASSP, pp. 4621–4624.

Yamagishi, J., 2006. Average-Voice-Based Speech Synthesis. Ph.D. Thesis, Tokyo Institute of Technology.

Yamagishi, J., Kobayashi, T., 2007. Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training. IEICE Trans. Inform. Systems E90-D (2), 533–543.

Yamagishi, J., Ling, Z.-H., King, S., 2008a. Robustness of HMM-based speech synthesis. In: Proc. Interspeech, pp. 581–584.

Yamagishi, J., Nose, T., Zen, H., Toda, T., Tokuda, K., 2008b. Performance evaluation of the speaker-independent HMM-based speech synthesis system "HTS-2007" for the Blizzard Challenge 2007. In: Proc. ICASSP, pp. 3957–3960.

Yamagishi, J., Zen, H., Wu, Y.-J., Toda, T., Tokuda, K., 2008c. The HTS-2008 system: yet another evaluation of the speaker-adaptive HMM-based speech synthesis system in the 2008 Blizzard Challenge. In: Proc. Blizzard Challenge Workshop.

Yamagishi, J., Kobayashi, T., Nakano, Y., Ogata, K., Isogai, J., 2009. Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm. IEEE Trans. Audio Speech Language Process. 17 (1), 66–83.

Yang, J.-H., Zhao, Z.-W., Jiang, Y., Hu, G.-P., Wu, X.-R., 2006. Multi-tier non-uniform unit selection for corpus-based speech synthesis. In: Proc. Blizzard Challenge Workshop.

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 1997. Speaker interpolation in HMM-based speech synthesis system. In: Proc. Eurospeech, pp. 2523–2526.

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 1998. Duration modeling for HMM-based speech synthesis. In: Proc. ICSLP, pp. 29–32.

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

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 2001. Mixed excitation for HMM-based speech synthesis. In: Proc. Eurospeech, pp. 2263–2266.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X.-Y., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P., 2006. The Hidden Markov Model Toolkit (HTK) Version 3.4. <http://htk.eng.cam.ac.uk/>.

Yu, J., Zhang, M., Tao, J., Wang, X., 2007. A novel HMM-based TTS system using both continuous HMMs and discrete HMMs. In: Proc. ICASSP, pp. 709–712.

Yu, Z.-P., Wu, Y.-J., Zen, H., Nankaku, Y., Tokuda, K., 2008. Analysis of stream-dependent tying structure for HMM-based speech synthesis. In: Proc. ICSP.

Yu, K., Toda, T., Gasic, M., Keizer, S., Mairesse, F., Thomson, B., Young, S., 2009. Probabilistic modelling of F0 in unvoiced regions in HMM based speech synthesis. In: Proc. ICASSP, pp. 3773–3776.

Zen, H., Lu, J., Ni, J., Tokuda, K., Kawai, H., 2003a. HMM-based prosody modeling and synthesis for Japanese and Chinese speech synthesis. Tech. Rep. TR-SLT-0032, ATR-SLT (in Japanese).

Zen, H., Tokuda, K., Kitamura, T., 2003b. Decision tree based simultaneous clustering of phonetic contexts, dimensions, and state positions for acoustic modeling. In: Proc. Eurospeech, pp. 3189–3192.

Zen, H., Nankaku, Y., Tokuda, K., Kitamura, T., 2006a. Speaker adaptation of trajectory HMMs using feature-space MLLR. In: Proc. Interspeech, pp. 2274–2277.

Zen, H., Toda, T., Tokuda, K., 2006b. The Nitech-NAIST HMM-based speech synthesis system for the Blizzard Challenge 2006. In: Proc. Blizzard Challenge Workshop.

Zen, H., Tokuda, K., Kitamura, T., 2006c. Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences. Comput. Speech Language 21 (1), 153–173.

Zen, H., Nankaku, Y., Tokuda, K., 2007a. Model-space MLLR for trajectory HMMs. In: Proc. Interspeech, pp. 2065–2068.

Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A., Tokuda, K., 2007b. The HMM-based speech synthesis system version 2.0. In: Proc. ISCA SSW6, pp. 294–299.

Zen, H., Toda, T., Nakamura, M., Tokuda, T., 2007c. Details of the Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005. IEICE Trans. Inform. Systems E90-D (1), 325–333.

Zen, H., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 2007d. A hidden semi-Markov model-based speech synthesis system. IEICE Trans. Inform. Systems E90-D (5), 825–834.

Zen, H., Nankaku, Y., Tokuda, K., 2008. Probabilistic feature mapping based on trajectory HMMs. In: Proc. Interspeech, pp. 1068–1071.

Zhang, L., 2009. Modelling Speech Dynamics with Trajectory-HMMs. Ph.D. Thesis, University of Edinburgh.