Temporal modeling in neural network based statistical parametric speech synthesis

Keiichi Tokuda, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku

Nagoya Institute of Technology

Abstract

This paper proposes a novel neural network structure for speech synthesis, in which spectrum, F0 and duration parameters are simultaneously modeled in a unified framework. In the conventional neural network approaches, spectrum and F0 parameters are predicted by neural networks while phone and/or state durations are given from other external duration predictors. In order to consistently model not only spectrum and F0 parameters but also durations, we adopt a special type of mixture density network (MDN) structure, which models utterance level probability density functions conditioned on the corresponding input feature sequence. This is achieved by modeling the conditional probability distribution of utterance level output features, given input features, with a hidden semi-Markov model, where its parameters are generated using a neural network trained with a log likelihood-based loss function. Variations of the proposed neural network structure are also discussed. Subjective listening test results show that the proposed approach improves the naturalness of synthesized speech.

Index Terms: speech synthesis, neural network, statistical parametric speech synthesis

1. Introduction

Hidden Markov model (HMM)-based speech synthesis [1] is one of the most popular approaches to statistical parametric speech synthesis [2], in which spectrum, Fundamental frequency (F0) and duration parameters are modeled in a unified framework [3] of Hidden semi-Markov model (HSMM), a special type of HMM with an explicit state duration modeling structure [4]-[7]. On the other hand, it has been shown that the neural network approach to statistical parametric speech synthesis [8] has a potential to improve the performance of statistical parametric speech synthesis systems [9, 10, 11]. However, temporal structures of speech are often modeled by using external duration predictors, *i.e.*, phone and/or state alignments are given by using some HMM-based system, and then phone and/or state durations are modeled by some duration predictor, which can be built by a separate neural network.

This paper describes a novel neural network-based speech synthesis approach for simultaneously modeling spectrum, F0 and duration in a unified framework. In order to consistently model not only spectrum and F0 parameters but also durations, we adopt a special type of mixture density network (MDN) [12] structure, which models utterance level probability density functions conditioned on the corresponding input feature sequence. This is achieved by modeling the conditional probability distribution of utterance level output features given input features with a hidden semi-Markov model (HSMM) [4]-[7], where its parameters are generated using a neural network trained with a log likelihood-based loss function. An efficient training algorithm based on this framework is derived, and the variations of the structure of the mixture density network is also discussed.

The rest of the paper is organized as follows. Section 2 defines the proposed neural network structure and then gives the training algorithm. Section 3 discusses variations of the proposed neural network structure. Preliminary experimental results are presented in Section 4. Concluding remarks are given in the final section.

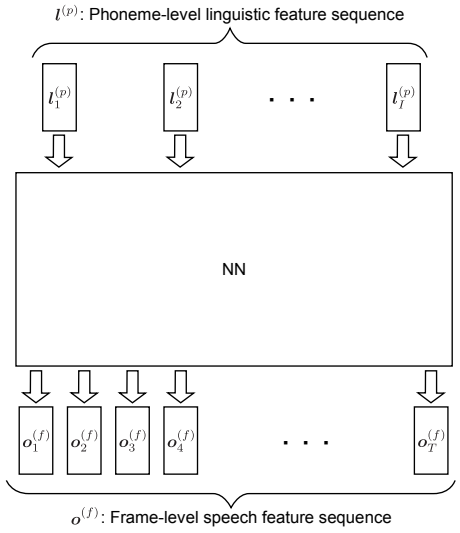

Figure 1: The basic problem of neural network based speech synthesis.

2. Temporal modeling based on neural networks

2.1. Basic problem of neural network based speech synthesis

The basic problem of neural network based speech synthesis is to convert the linguistic feature sequence

$$\boldsymbol{l}^{(p)} = \left\{\boldsymbol{l}_1^{(p)}, \boldsymbol{l}_2^{(p)}, \ldots, \boldsymbol{l}_I^{(p)}\right\}, \tag{1}$$

which corresponds to an utterance, to a speech feature sequence

$$\boldsymbol{o}^{(f)} = \left\{\boldsymbol{o}_1^{(f)}, \boldsymbol{o}_2^{(f)}, \ldots, \boldsymbol{o}_T^{(f)}\right\} \tag{2}$$

by a neural network as illustrated in Fig. 1, where I is the number of phonemes and T is the number of frames in the utterance.

To avoid the inconvenience that T and I are changed utterance-by-utterance, the frame-by-frame structure shown in

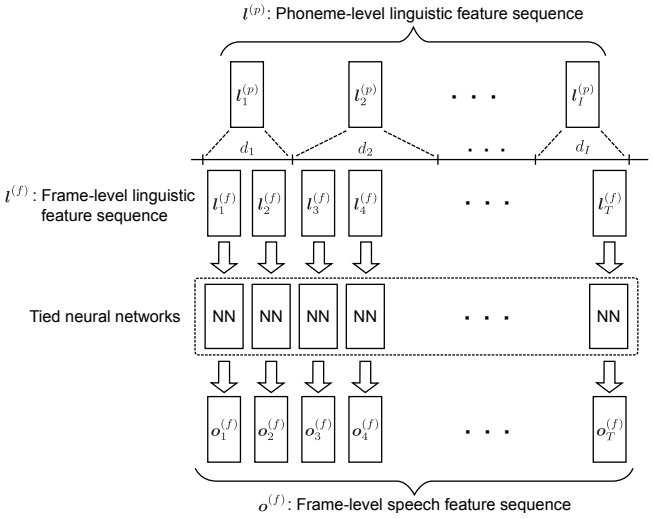

Figure 2: Frame-by-frame conversion. It needs an external duration predictor to determine phoneme durations $\left\{d_1^{(p)}, d_2^{(p)}, \ldots, d_I^{(p)}\right\}$.

Fig. 2 is often used [9]. In this case, we need an external duration predictor since the phoneme-level sequence $\boldsymbol{l}^{(p)}$ has to be *upsampled* to a frame-level sequence $\boldsymbol{l}^{(f)}$. It should be noted that $\boldsymbol{l}^{(f)}$ can include phoneme duration information and positional information in each phoneme.

2.2. Proposed neural network structure

To solve the above-mentioned problem, we derive a unified neural network structure based on the MDN [12] framework. MDNs give probability density functions over output features conditioned on the corresponding input features, by modeling the conditional probability distribution of output features given input features with a Gaussian mixture model (GMM), where its parameters are generated from an neural network trained with a log likelihood-based loss function.

The conventional MDN-based speech synthesis system [11] is shown in Fig. 3. As described in [11], the frame-level linguistic feature sequence $\boldsymbol{l}^{(f)}$ include phoneme duration information and positional information in each phoneme given from an external duration predictor, and the MDN defines frame-level likelihoods of speech features, *i.e.*, spectrum and F0 parameters.

On the other hand, the proposed system uses the HSMM structure, instead of GMM structure for defining the utterance-level likelihood

$$p(\boldsymbol{o}^{(f)} \mid \boldsymbol{\lambda}^{(s)}) = \sum_{\boldsymbol{q}} \left\{ \prod_{t=1}^{T} p(\boldsymbol{c}_t \mid \boldsymbol{\mu}_{q_t}, \boldsymbol{\Sigma}_{q_t}) \prod_{k=1}^{K} p(d_k \mid \xi_k, \sigma_k^2) \right\} \tag{3}$$

as shown in Fig. 4, where

$$\boldsymbol{q} = \{q_1, q_2, \ldots, q_T\} \tag{4}$$

$$= \{ \underbrace{1, \ldots, 1}_{d_1}, \underbrace{2, \ldots, 2}_{d_2}, \ldots\ldots, \underbrace{K, \ldots, K}_{d_K} \}. \tag{5}$$

The linguistic feature sequence

$$\boldsymbol{l}^{(s)} = \left\{ \boldsymbol{l}_1^{(s)}, \boldsymbol{l}_2^{(s)}, \ldots, \boldsymbol{l}_K^{(s)} \right\} \tag{6}$$

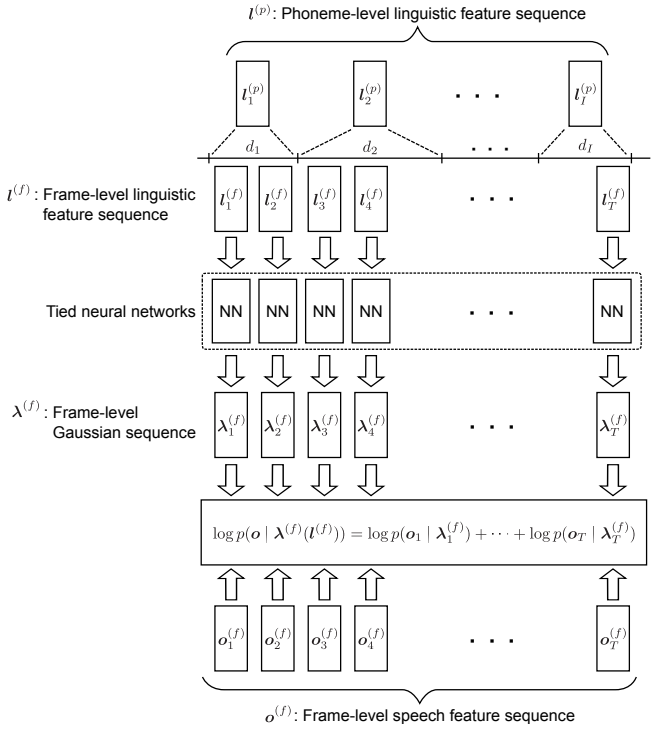

Figure 3: The conventional MDN-based speech synthesis system. It still needs an external duration predictor.

is fed into the neural network in a state-by-state manner, and the neural network outputs state Gaussians for speech features,[1] $\mathcal{N}(\boldsymbol{o}_t \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, and Gaussians for state durations, $\mathcal{N}(d_k \mid \xi_k, \sigma_k^2)$, *i.e.*,

$$\boldsymbol{\lambda}^{(s)} = \left\{ \boldsymbol{\lambda}_1^{(s)}, \boldsymbol{\lambda}_2^{(s)}, \ldots, \boldsymbol{\lambda}_K^{(s)} \right\}, \tag{7}$$

where

$$\boldsymbol{\lambda}_k^{(s)} = \left\{ (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), (\xi_k, \sigma_k^2) \right\}. \tag{8}$$

In Fig. 4, $\boldsymbol{\lambda}_{i,j}^{(s)}$ is defined as

$$\boldsymbol{\lambda}_{i,j}^{(s)} = \boldsymbol{\lambda}_k^{(s)} \Big|_{k=(i-1)\cdot J+j}, \tag{9}$$

where i is the phoneme index, j is the state index in each phoneme, I is the number of phonemes in an utterance, J is the number of states in each phoneme, and K is the number of states in the utterance, *i.e.*, $K = I \cdot J$,

By using HSMM as an objective function in passing the error back, we can minimize the error based on temporal alignment. It should be noted that the HSMM structure defining the utterance-level likelihoods also works as a rate-converter from state-level to frame-level. It is also noted that phoneme- to state-level conversion ($\boldsymbol{l}^{(p)} \rightarrow \boldsymbol{l}^{(s)}$) is straightforward because each phoneme always consists of a fixed number of states; in the experiment in Section 4, we had five states in each phoneme, *i.e.*, $J = 5$.

[1] The speech feature vector consists of spectrum and F_0 parameters, and the F_0 part has multi-space probability distributions. However, for simplicity of notation, we represent them as simple Gaussians.

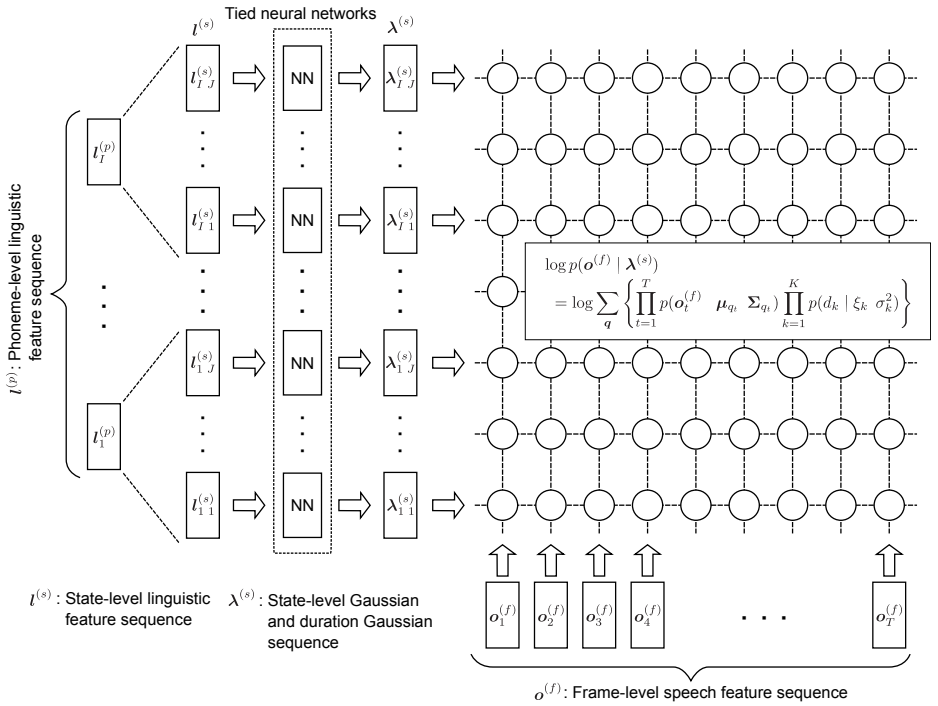

Figure 4: The proposed neural network based speech synthesis system. It can model temporal structure of speech in a unified framework.

2.3. Training algorithm

The auxiliary function of the likelihood function Eq. (3) can be written as

$$\mathcal{Q}(\overline{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)}) = \sum_{\boldsymbol{q}} p(\boldsymbol{q} \mid \boldsymbol{o}, \bar{\boldsymbol{\lambda}}^{(s)}) \log p(\boldsymbol{o}, \boldsymbol{q} \mid \boldsymbol{\lambda}^{(s)}). \quad (10)$$

By using the following probabilities calculated by the generalized forward-backward algorithm as

$$\begin{aligned}\alpha_t(j) =& p(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t, q_t = j \mid q_{t+1} \neq j, \bar{\boldsymbol{\lambda}}^{(s)}) \\ =& \sum_{d=1}^{t} \sum_{i=1, i\neq j}^{K} \alpha_{t-d}(i) p(d \mid \bar{\xi}_j, \bar{\sigma}_j^2) \prod_{s=t-d+1}^{t} p(\boldsymbol{o}_s \mid \bar{\boldsymbol{\mu}}_j, \bar{\boldsymbol{\Sigma}}_j),\end{aligned} \quad (11)$$

$$\begin{aligned}\beta_t(i) =& p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T q_t = i \mid q_t \neq i, \bar{\boldsymbol{\lambda}}^{(s)}) \\ =& \sum_{d=1}^{T-t} \sum_{j=1, j\neq i}^{K} p(d \mid \bar{\xi}_j, \bar{\sigma}_j^2) \prod_{s=t+1}^{t+d} p(\boldsymbol{o}_{t+1} \mid \bar{\boldsymbol{\mu}}_j, \bar{\boldsymbol{\Sigma}}_j) \beta_{t+d}(j),\end{aligned} \quad (12)$$

$$\begin{aligned}\gamma_j(t) =& \frac{1}{p(\boldsymbol{o} \mid \bar{\boldsymbol{\lambda}})} \sum_{t_0=1}^{t-1} \sum_{t_1=t}^{T} \sum_{i\neq j} \alpha_{t_0}(i) \\ & \times \prod_{s=t_0+1}^{t_1} p(\boldsymbol{o}_s \mid \bar{\boldsymbol{\mu}}_j, \bar{\boldsymbol{\Sigma}}_j) p(t_1 - t_0 \mid \bar{\xi}_j, \bar{\sigma}_j^2) \beta_{t_1}(j),\end{aligned} \quad (13)$$

$$\begin{aligned}\chi_j(d) =& \frac{1}{p(\boldsymbol{o} \mid \bar{\boldsymbol{\lambda}})} \sum_{t=d}^{T} \sum_{i\neq j} \alpha_{t-d}(i) \\ & \times \prod_{s=t-d+1}^{t} p(\boldsymbol{o}_s \mid \bar{\boldsymbol{\mu}}_j, \bar{\boldsymbol{\Sigma}}_j) p(d \mid \bar{\xi}_j, \bar{\sigma}_j^2) \beta_t(j)\end{aligned} \quad (14)$$

the partial derivatives of Eq. (10) w.r.t $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \xi_k, \sigma_k^2$ can be derived as follows:

$$\frac{\partial \mathcal{Q}(\overline{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)})}{\partial \boldsymbol{\mu}_k} = \sum_{t=1}^{T} \gamma_k(t) \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{o}_t - \boldsymbol{\mu}_k), \quad (15)$$

$$\frac{\partial \mathcal{Q}(\overline{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)})}{\partial \boldsymbol{\Sigma}_k^{-1}} = \frac{1}{2} \sum_{t=1}^{T} \gamma_k(t) \left(\boldsymbol{\Sigma}_k - (\boldsymbol{o}_t - \boldsymbol{\mu}_k)(\boldsymbol{o}_t - \boldsymbol{\mu}_k)^\top \right), \quad (16)$$

$$\frac{\partial \mathcal{Q}(\overline{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)})}{\partial \xi_k} = \sum_{d=1}^{D} \chi_k(d) \sigma_k^{-2} (d - \xi_k), \quad (17)$$

$$\frac{\partial \mathcal{Q}(\overline{\boldsymbol{\lambda}}^{(s)}, \boldsymbol{\lambda}^{(s)})}{\partial \sigma_k^{-2}} = \frac{1}{2} \sum_{d=1}^{D} \chi_k(d) \left(\sigma_k^2 - (d - \xi_k)^2 \right). \quad (18)$$

Thus, by back-propagating the derivatives of the log likelihood function through the network, the neural network weights can be updated to maximize the log likelihood.

In the conventional neural network based systems often use HMM-based state or phone alignments to train external duration predictors. In the proposed framework, a unified neural network is trained for modeling speech features and state or phoneme durations simultaneously. Thus, it is expected to reach a totally-optimal solution, and result in a better performance.

3. Other choices of model structures

3.1. Phoneme-level implementation

In the model structure described in Section 2, the neural network runs at the state-level. It can also run at the phoneme-level as shown in Fig. 5. In this case, the input

$$\boldsymbol{l}^{(p)} = \left\{ \boldsymbol{l}_1^{(p)}, \boldsymbol{l}_2^{(p)}, \ldots, \boldsymbol{l}_I^{(p)} \right\} \quad (19)$$

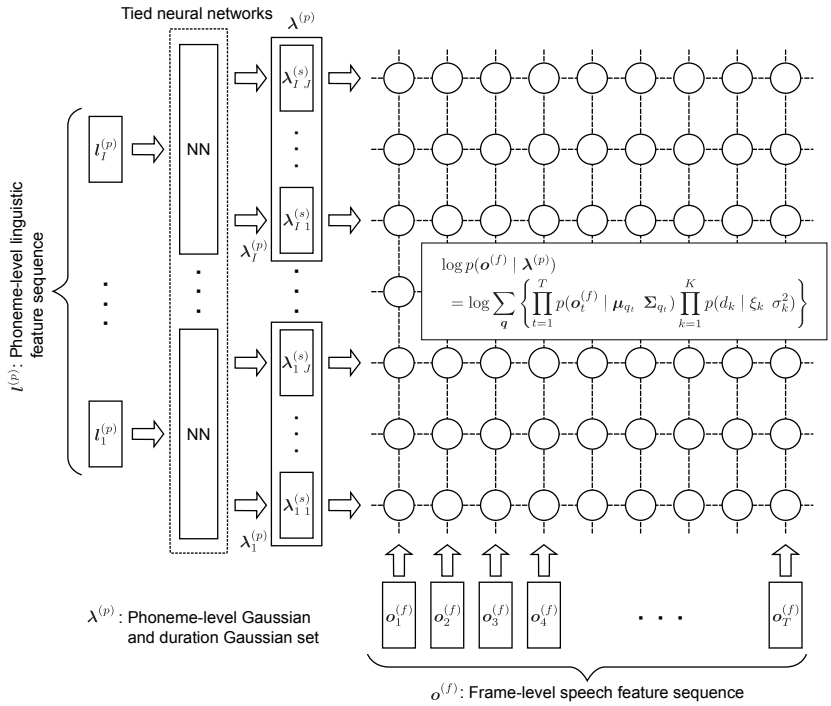

Figure 5: The proposed MDN-based speech synthesis system, which runs at the phoneme-level.

and output

$$\boldsymbol{\lambda}^{(p)} = \left\{ \boldsymbol{\lambda}_1^{(p)}, \boldsymbol{\lambda}_2^{(p)}, \ldots, \boldsymbol{\lambda}_I^{(p)} \right\}, \tag{20}$$

of the neural network are in the phoneme-level, where $\boldsymbol{\lambda}_i^{(p)}$ is given as

$$\boldsymbol{\lambda}_i^{(p)} = \left\{ \boldsymbol{\lambda}_{i,1}^{(s)}, \boldsymbol{\lambda}_{i,2}^{(s)}, \ldots, \boldsymbol{\lambda}_{i,J}^{(s)} \right\}. \tag{21}$$

It should be noted that the proposed framework may drastically reduces the computational cost in both training and synthesis because the neural network runs only at each state or each phoneme. When we assume a speaking rate of ten phonemes per second and a 5-ms frame shift, and phoneme models with 5-state topology, we compare computational costs as follows:

Frame-level:	200 times per second
State-level:	50 times per second
Phoneme-level:	10 times per second

If the neural network has the same size for all three cases, computational costs of "state-level" and "phoneme-level" are almost $1/4$ and $1/20$, respectively, of "frame-level."

3.2. Untying state Gaussians of HSMM

Each state of the HSMM can be expanded as shown in Fig. 6 (a), in which all Gaussians corresponding to a state are tied together. By untying those Gaussians, we can derive other types of model structures as shown in Fig. 6 (b) and (c). It is noted that the structure in Fig. 6 (c) corresponds to an acoustic model described in [13].

3.3. Discrete probability distributions for duration modeling

In the above discussion, we assumes continuous probability distributions for duration modeling, while discrete probability distributions can also be used. In this case, for modeling state or phoneme durations as discrete probability distributions, a softmax function is used in a corresponding part of the output layer of the neural network.

4. Experiment

4.1. Experimental conditions

Japanese 503 utterances, which can be downloaded from HTS web site[2], were used in these experiments. The contents of the data were the same as the B-set of the ATR phonetically balanced Japanese speech database [14]. The 450 utterances were used for training and the remaining 53 utterances were used for testing. Speech signals were sampled at 48 kHz. Feature vectors were extracted with a 5-ms shift and the feature vector consisted of the 0-th through 49-th mel-cepstral coefficients and a log F_0 value. Mel-cepstral coefficients were extracted from the smoothed spectrum analyzed by the STRAIGHT [15].

In these experiments, four systems were compared.

- **HSMM**: Conventional HMM-based speech synthesis system
- **MDN-frame-phoneme**: Speech synthesis based on frame-level MDN with phoneme durations obtained from natural speech
- **MDN-frame-state**: Speech synthesis based on frame-level MDN with state durations obtained from natural speech
- **MDN-state-state**: Speech synthesis based on state-level MDN with state durations obtained from natural speech
- **MDN-HSMM**: Speech synthesis based on state-level MDN which outputs state Gaussian and duration Gaussian (Fig. 4)

[2] http://hts.sp.nitech.ac.jp/

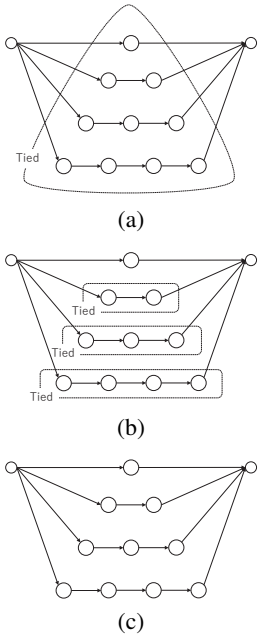

Figure 6: Untying state Gaussians in the HSMM: (a) expanded topology of the HSMM; (b) state duration-dependent Gaussians; (c) State duration-dependent sequences of Gaussians.

In **HSMM**, five-state, left-to-right, no-skip hidden semi-Markov models (HSMMs) were used. HSMMs modeled observation vectors consisting of 50 mel-cepstral coefficients, log F_0 values, and their dynamic features (delta and delta-delta). To model log F_0 sequences consisting of voiced and unvoiced observations, a multi-space probability distribution (MSD) was used. The minimum description length (MDL) criterion was employed to determine the size of decision tree for context clustering [16]. The input feature for **MDN-frame-phoneme** was a 411-dimensional feature vector, consisting of 408 linguistic features including binary features and numerical features for contexts and three duration features including duration of the current phoneme and the relative position of the current frame in the phoneme. The input feature for **MDN-frame-state** consisted of 408 linguistic features, three duration features, including duration of the current state and the relative position of the current frame in the state, and five binary features representing the state index in the phoneme. For **MDN-state-state** and **MDN-HSMM**, 408 linguistic features and five binary features representing the state index in the phoneme were used as the input feature. The input features were normalized to be within 0.0-1.0 based on their minimum and maximum values in the training data. The acoustic feature vector for MDN-based systems consisted of 50 mel-cepstral coefficients and a log F_0 value, which were normalized to have zero-mean unit-variance, their dynamic features (delta and delta-delta), and a voiced/unvoiced binary value. The input and acoustic feature vectors for **MDN-frame-phoneme**, **MDN-frame-state**, and **MDN-state-state** were time-aligned frame-by-frame by **HSMM**. A single network which modeled both spectral and excitation parameters was trained. The architecture of the MDNs used in **MDN-frame-phoneme**, **MDN-frame-state**, **MDN-state-state**, and **MDN-HSMM** was 3-hidden-layer with 1024 units per layer. The sigmoid activation function was used in the hidden layers and the linear activation function

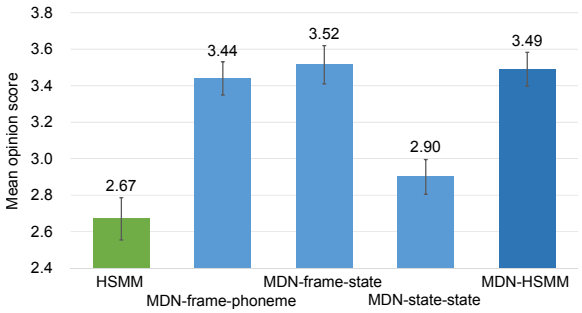

Figure 7: Mean opinion scores of the five speech synthesis systems. MDN-frame-phoneme, MDN-frame-state, and MDN-state-state are using durations obtained from the alignment to natural speech, while HSMM and MDN-HSMM are using predicted durations.

was used in the output layer. The weights of the MDN were initialized randomly, then they were optimized by maximizing likelihood. For training the MDNs, a mini-batch stochastic gradient descent (SGD)-based back-propagation algorithm was used. For **MDN-HSMM**, one utterance was used as one mini-batch in SGD-based training. The basic parameter generation algorithm was applied to generate parameter trajectories for all systems. The duration information for test data was derived from forced-alignment to natural speech with **HSMM** for **MDN-frame-phoneme**, **MDN-frame-state**, and **MDN-state-state**. In **HSMM** and **MDN-HSMM**, the durations for each state were predicted by each duration model.

4.2. Experimental results

To evaluate the naturalness of the synthesized speech, a subjective listening test was conducted. The naturalness of the synthesized speech was assessed by the mean opinion score (MOS) test method. The subjects were ten Japanese students in our research group. Twenty sentences were chosen at random from the test sentences. Speech samples were presented in random order for each test sentence. In the MOS test, after listening to each test sample, the subjects were asked to assign the sample a five-point naturalness score (5: natural – 1: poor).

Figure 7 shows the subjective evaluation results. From this figure, it can be seen that the MDN-based systems show significantly improvement from the HMM-based system **HSMM**. This result indicates the effectiveness of the use of MDNs for modeling acoustic features. Comparing **MDN-frame-phoneme** with **MDN-frame-state**, **MDN-frame-state** show the better result than **MDN-frame-phoneme**. Thus, the state information is useful to improve the naturalness of synthesized speech. Comparing **MDN-frame-state** with **MDN-state-state**, **MDN-state-state** degrade the naturalness of synthesized speech from **MDN-frame-state**. This result indicates that the duration information and the position in the current state is useful to model acoustic features. However, **MDN-HSMM** shows significant improvement from MDN-state-state though **MDN-HSMM** was based on state-level modeling as MDN-state-state. Additionally, **MDN-HSMM** has comparable performance to **MDN-frame-state** though **MDN-HSMM** used the predicted durations rather than the duration obtained from the alignment to natural speech. These results suggest that the utterance-level training algorithm such as forward-backward algorithm is effective to model utterances with MDNs.

5. Conclusions

We have presented a novel speech synthesis approach, in which not only spectrum and F0 but also duration parameters are simultaneously modeled in a unified framework of neural networks. Its training algorithm based on the generalized forward-backward algorithm was derived. We also showed variations of the model structures and discussed their advantages and disadvantages. A preliminary experimental result based on a listening test showed that the proposed approach improved the naturalness of the synthesize speech. Future work includes conducting a large scale listening test to compare variations of model structures presented in Section 3. Combining the proposed approach and an approach to directly modeling speech waveforms [17] is also included in our future work.

6. Acknowledgements

This research was partly funded by Core Research for Evolutionary Science and Technology (CREST) from Japan Science and Technology Agency (JST).

7. References

[1] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proc. IEEE*, vol. 101, no. 5, pp. 1234–1252, 2013.

[2] H. Zen, K. Tokuda, and A. Black, "Statistical parametric speech synthesis," *Speech Commn.*, vol. 51, no. 11, pp. 1039–1064, 2009.

[3] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," *IEICE Trans. Inf. Syst.*, vol. E90-D, no. 5, pp. 825–834, 2007.

[4] J. Ferguson, "Variable duration models for speech," in *Proc. the Symposium on the Application Hidden Markov Models to Text and Speech*, 1980, pp. 143–179.

[5] M. Russell and R. Moore, "Explicit modeling of state occupancy in hidden Markov models for automatic speech recognition," in *Proc. ICASSP*, 1985, pp. 5–8.

[6] S. Levinson, "Continuously variable duration hidden Markov models for automatic speech recognition," *Comput. Speech Lang.*, vol. 1, pp. 29–45, 1986.

[7] C. Mitchell, M. Harper, and L. Jamieson, "On the complexity of explicit duration HMMs," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 3, pp. 213–217, 1995.

[8] Z.-H. Ling, S. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 35–52, 2015.

[9] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. ICASSP*, 2013, pp. 7962–7966.

[10] Y. Qian, Y. Fan, W. Hu, and F. Soong, "On the training aspects of deep neural network (DNN) for parametric TTS synthesis," in *Proc. ICASSP*, 2014, pp. 3857–3861.

[11] H. Zen and A. Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis," in *Proc. ICASSP*, 2014, pp. 3872–3876.

[12] C. Bishop, "Mixture density networks," Neural Computing Research Group, Aston University, Tech. Rep. NCRG/94/004, 1994.

[13] R. Terashima, H. Zen, Y. Nankaku, and K. Tokuda, "A frame-based context-dependent acoustic modeling for speech recognition," *IEEJ Transactions on Electronics, Information and Systems*, vol. 130, no. 10, pp. 1856–1864, 2010. [Online]. Available: http://ci.nii.ac.jp/naid/130000415125/en/

[14] A. Kurematsu, K. Takeda, Y. Sagisaka, H. Katagiri, S. Kuwabara, and K. Shikano, "ATR Japanese speech database as a tool of speech recognition and synthesis," *Speech Communication*, vol. 9, pp. 257–285, 1990.

[15] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based f_0 extraction: possible role of a repetitive structure in sounds," *Speech Commn.*, vol. 27, pp. 187–207, 1999.

[16] K. Shinoda and T. Watanabe, "Acoustic modeling based on the MDL criterion for speech recognition," in *Proc. Eurospeech*, 1997, pp. 99–102.

[17] K. Tokuda and H. Zen, "Directly modeling speech waveforms by neural networks for statistical parametric speech synthesis," in *Proc. ICASSP*, 2015, pp. 4215–4219.