INITIAL INVESTIGATION OF SPEECH SYNTHESIS BASED ON COMPLEX-VALUED NEURAL NETWORKS

Qiong Hu[1], *Junichi Yamagishi*[1], *Korin Richmond*[1], *Kartick Subramanian*[2], *Yannis Stylianou*[3]

[1]The Centre for Speech Technology Research, University of Edinburgh, UK
[2]School of Computer Engineering, Nanyang Technological University, Singapore
[3]Toshiba Research Europe Ltd, Cambridge, U.K.

ABSTRACT

Although frequency analysis often leads us to a speech signal in the complex domain, the acoustic models we frequently use are designed for real-valued data. Phase is usually ignored or modelled separately from spectral amplitude. Here, we propose a complex-valued neural network (CVNN) for directly modelling the results of the frequency analysis in the complex domain (such as the complex amplitude). We also introduce a phase encoding technique to map real-valued data (e.g. cepstra or log amplitudes) into the complex domain so we can use the same CVNN processing seamlessly. In this paper, a fully complex-valued neural network, namely a neural network where all of the weight matrices, activation functions and learning algorithms are in the complex domain, is applied for speech synthesis. Results show its ability to model both complex-valued and real-valued data.

Index Terms**—** complex-valued neural network, speech synthesis, complex amplitude, phase modelling

1. INTRODUCTION

For many real-valued signals (e.g. image or audio), one of the most frequently used approaches is frequency-domain analysis such as the Fourier transform, which normally leads us to a single $z \in C$ in an Euler representation of the complex domain, $z = Ae^{i*\varphi} = A(\cos\varphi + i \sin\varphi)$ where $A \in R$ and $\varphi \in R$ are the amplitude and phase of the signal respectively. The analysis of the amplitudes of each frequency bin, that is, spectral amplitude analysis, is dominant in many speech processing applications because of its relevance to speech perception. Its parameterisations using cepstra, line spectral pairs or log amplitudes are well studied for the statistical modelling used in speech synthesis. Various models have been proposed to model the statistical behaviour of these parameters, e.g. hidden Markov models (HMM) [1], deep neural networks (DNN) [2] and linear dynamic models (LDM) [3].

Meanwhile, recent studies have elaborated the potential of using phase features in speech enhancement [4], recognition [5] and synthesis [6]. The common strategy among these methods is to analyse and model the amplitude and phase separately. There have been various attempts at phase representation, e.g. relative phase shift [7], group delay [8], phase dispersion [9], phase distortion [10] and the complex cepstrum [6] for speech synthesis. For example, in [6] and [11], complex cepstra or a cepstrum-like representation calculated from the standard deviation of phase distortion have been modelled, respectively, using an additional independent stream in HMM-based statistical parametric speech synthesis (SPSS) to improve the quality of the vocoded speech.

An alternative approach to such explicit and separate amplitude and phase feature representations is to combine amplitude and phase together by representing a signal as a complex value $z = u + iv \in C$ and then modelling the signal z using a new statistical model that can deal with complex numbers directly. Here, we may use both the amplitude and phase information of the signal as a part of the new objective function in the complex domain $E_C(z) = \hat{E}_C(A, \varphi)$ for training the models so that the model can consider errors of the amplitude A and phase φ of the signal z jointly. There are a few examples of pioneering work we can look at that have extended statistical models into the complex domain. [12] has defined a "complex normal distribution" using a mean vector, covariance and relation matrices, which is a normal distribution in the complex domain. Although there is little work in the literature about how to define HMMs for complex-value observations, there are a few nice attempts to extend neural networks into the complex domain, which is referred to as a "complex-valued neural network" (CVNN) [13, 14, 15, 16, 17]. Since DNNs, which use many stacked layers, have shown their effectiveness for improving the quality of synthetic speech, it is theoretically and scientifically interesting to extend the neural network-based speech synthesis into the CVNN framework.

The first approach to modelling complex-valued signals using CVNNs was to split real and imaginary parts of the complex-valued signals into two real-valued signals [13] and to use normal real-valued neutral networks. However, this resulted in a poor approximation, especially of phase, because this model cannot represent relationships between real and imaginary parts properly, and hence the gradient to be used for model training may be incorrect. Therefore, a so-called fully CVNN where all inputs/outputs, weight matrices and activation functions are in the complex domain, along with

This work was partially supported by Toshiba Research Europe Ltd, Cambridge, UK. Part of this work was also supported by EPSRC through Programme Grant EP/I031022/1 (NST) and EP/J002526/1 (CAF)

a corresponding training algorithm have been proposed by [14, 15, 16]. It has already been applied to wind prediction, imagine enhancement, and landmine prediction [17] and has shown its effectiveness. However as far as we know, its application to speech synthesis has not been reported yet.

So, in this paper, we investigate a few ways to apply a full CVNN to SPSS. The technical challenges we face are: i) In the literature, CVNNs where input and output vectors are complex-valued have mainly been investigated, whereas for SPSS, linguistic vectors are real-valued; ii) In addition to cases where acoustic features are complex-valued, it is also interesting to apply the CVNN into the traditional real-valued acoustic features. This is motivated by the fact that for real-valued classification tasks, a CVNN has the same performance as a real-valued NN with a larger number of neurons [18]. Note that speech synthesis is a regression task, which is different from tasks reported in the literature; iii) Complex amplitudes extracted from [19] can be used as complex-valued outputs where phase is composed of linear phase, minimum phase and disperse phase. Here, linear phase should be omitted in the calculation of the amplitude-phase objective function since analysis window position is unrelated to linguistic input. However, the computation of disperse phase from a sparse sinusoid representation has not been studied yet. Hence, its inaccurate calculation may affect the analysis of the CVNN's ability to model complex-valued acoustic features.

Therefore, in this preliminary work complex amplitudes with only calculated minimum phase are considered for the fully complex-valued feed-forward network. Its activation functions have to be carefully defined so its differentiable gradients exist almost everywhere in the complex plane. In this paper, a complex exponential function, which has singularity points at $\pm\infty$ only is used at the output layer. For the learning algorithm, a complex-valued back-propagation algorithm using a logarithmic minimisation criterion which includes both amplitude and phase errors is used. We also apply the CVNN to model real acoustic features for SPSS. For this purpose, a phase encoding technique is introduced to map the real-valued data into the complex domain.

This paper is organised as follows. Since CVNNs are not generally known in the synthesis field, we overview them in Section 2. Our experiments are reported in Section 3. We then summarise this preliminary work in Section 4.

2. COMPLEX-VALUED NEURAL NETWORKS

2.1. CVNN architecture

Here we explain CVNN formulations using a one hidden layer network as an example. Deeper architectures may also be constructed. Let $\boldsymbol{x} = [x_1, \cdots, x_m]^\top \in C^m$ and $\boldsymbol{y} = [y_1, \cdots, y_n]^\top \in C^n$ be the m-dimensional input and n-dimensional output complex-valued vectors for the network, respectively. A projection operation from the input layer to the hidden layer $\boldsymbol{z} = [z_1, \cdots, z_h] \in C^h$ using a complex-valued matrix $\boldsymbol{W}_{\text{in}} \in C^{h\times m}$ can be written as:

$$\boldsymbol{z} = [z_1, \cdots, z_h] = \boldsymbol{f}_C(\boldsymbol{W}_{\text{in}}\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{f}_C(\cdot)$ denotes an element-wise complex-valued non-linear activation operation and each element is transformed using $f_C(z)$. Then a linear projection operation from the hidden layer to the output layer using a complex-valued matrix $\boldsymbol{W}_{out} \in C^{n\times h}$ can also be written as:

$$\boldsymbol{y} = \boldsymbol{W}_{\text{out}}\boldsymbol{z}. \tag{2}$$

So the CVNN architecture is almost the same as normal neural networks apart from the complex-valued non-linear activation function $f_C(z)$, which is described in the next section.

2.2. Complex-valued activation function

As all the inputs and weights in a CVNN are complex-valued, the activation function also has to be extended into the complex domain. The complex activation function should be "almost bounded" and differentiable according to Liouville's theorem [20] so that we can derive the gradient based back-propagation algorithm. In the classic approach [13], two real valued functions were used for real and imaginary parts separately as an approximated activation function $f_{C\to R}(z)$. An example of such a function is as follows: $f_C(z) \approx f_{C\to R}(z) = \sqrt{f_R(u)^2 + f_R(v)^2}$, where f_R is a normal-valued activation function such as a sigmoid function. Later a set of elementary transcendental functions such as "asinh", "atan", "atanh", "asin", "tan", or "tanh" [16], which have a limited number of singular points, were suggested as possible choices of activation functions for the full CVNN.

Recently, the complex version of an exponential function was proposed as a good activation function for the fully complex CVNN [21], as its singularities are located at $\pm\infty$ only, which ensures the activation function is continuous in the input range. The exponential function can also help avoid calculating the derivative ($\frac{1}{y}$) of the logarithmic error (Section 2.3) during back-propagation. Therefore, instead of a linear function, an exponential function is employed at the output layer. The complex version of an exponential function can be written as:

$$f_C(z) = f'_C(z) = e^{u+i*v} = e^u(\cos v + i \sin v). \tag{3}$$

2.3. Objective functions and back-propagation

The back-propagation algorithm, which calculates the gradient of an objective function $E_C(y, \widehat{y})$ with respect to all the weights in the CVNN, can also be clearly defined. Here $\widehat{\boldsymbol{y}} = [\widehat{y}_1, \cdots, \widehat{y}_n]^\top \in C^n$ denotes a target complex-valued vector and $\widehat{y} \in C$ denote an element of the vector. Mean squared error is often used as the minimisation criterion. For complex-valued signals, the squared error represents only the magnitude of error explicitly and does not include the phase error directly. Here, instead, a logarithmic error function [21],

Table 1. *Configuration for different systems*

ID	Spectral feature	Phase	System
INT-En-C	RDC	Encoded	CVNN
DIR-En-C	log amplitude	Encoded	CVNN
DIR-Ze-R	log amplitude	Zero	RVNN
CDIR-Mi-C	complex amplitude	Minimum	CVNN

which includes both magnitude and phase error explicitly is used as the objective function.

$$E_C(y,\widehat{y}) = \frac{1}{2}\left[\log\left[\frac{y}{\widehat{y}}\right]\overline{\log\left[\frac{y}{\widehat{y}}\right]}\right] \tag{4}$$

$$= \frac{1}{2}\left[\log\left[\frac{|y|}{|\widehat{y}|}\right]^2 + \left[Arg(y) - Arg(\widehat{y})\right]^2\right] \tag{5}$$

$$= \frac{1}{2}\left[\log\left[\frac{A_y}{A_{\widehat{y}}}\right]^2 + \left[\varphi_y - \varphi_{\widehat{y}}\right]^2\right] \tag{6}$$

where $\overline{\log\left[\frac{y}{\widehat{y}}\right]}$ is the complex-conjugate of $\log\left[\frac{y}{\widehat{y}}\right]$, A_y and $A_{\widehat{y}}$ are magnitudes of y and $\widehat{y}$, respectively and φ_y and $\varphi_{\widehat{y}}$ are phases of y and $\widehat{y}$, respectively. Moreover, constants k_1 and k_2 may further be introduced as weighting factors for the magnitude and phase errors:

$$E_C(y,\widehat{y}) = \frac{1}{2}\left[k_1\log\left[\frac{A_y}{A_{\widehat{y}}}\right]^2 + k_2\left[\varphi_y - \varphi_{\widehat{y}}\right]^2\right] \tag{7}$$

Based on the objective function, the derivative of the objective function with respect to the l-th row k-th column element of the output weight matrix $\boldsymbol{W}_{out}$ denoted $w_{lk} = w^R_{lk} + i * w^I_{lk} \in C$ is given by

$$\frac{\partial E_C(y_l,\widehat{y}_l)}{\partial w_{lk}} = \frac{\partial E_C(y_l,\widehat{y}_l)}{\partial w^R_{lk}} + i\frac{\partial E_C(y_l,\widehat{y}_l)}{\partial w^I_{lk}} \tag{8}$$

By chain rule application, the update of w_{lk}, that is Δw_{lk}, is given by

$$\Delta w_{lk} = \delta\,\overline{z_k}\left[k_1\log\left[\frac{A_y}{A_{\widehat{y}}}\right] + i * k_2\left[\varphi_y - \varphi_{\widehat{y}}\right]\right] \tag{9}$$

where $\overline{z}_k$ is the conjugate of k-th hidden unit of $\boldsymbol{z}$. δ is the learning rate [1]. For the derivation of the updates of $\boldsymbol{W}_{in}$, please refer to [20, 21].

2.4. Phase coding

For the special case where the input vector is real-valued, it is empirically recommended to transform it into the complex representation [15]. For this we adopt a heuristic solution called phase encoding [15] using the transformation $\tilde{x} = \cos x' + i\sin x'$ where $x' \in R$ is the real-valued data and $\tilde{x} \in C$ is the obtained complex value, which is located on the unit circle. Note that in order to ensure a one-to-one mapping, x' is normalised to lie within the circle beforehand.

3. EVALUATION FOR SPEECH SYNTHESIS

3.1. System configuration

Speech data [22] from a British male professional speaker is used for training the synthesis system. The database consists of 2400 utterances for training, 70 for testing, recorded with a sample rate of 16kHz. The input features consist of 160 bottleneck features [23] as a compact, learned linguistic representation. For spectral features, either i) 50 regularized discrete cepstra (RDC) extracted from the amplitudes of the harmonic dynamic model (HDM) [24] or ii) 50 highly correlated log amplitudes from the perceptual dynamic sinusoidal model (PDM) [25] are used as real-valued spectral output. 50 complex amplitudes with minimum phase extracted from PDM [19] are applied as complex-valued spectral output. Continuous $logF_0$ and a voiced/unvoiced (vuv) binary value together with either type of these spectral features are used to represent output features (total dimensions: 52). Maximum likelihood parameter generation [26] and slope information from the dynamic sinusoidal model are not included in this paper. Both real-valued inputs and outputs are normalised and then phase encoded by preprocessing. For complex amplitudes, only amplitude is normalised. For the CVNN systems, two hidden layers are used with 100 complex neurons per layer. $Sinh$ and $expotential$ function are used as hidden and output layer activation functions. The values of the weighting factors $k1$ and $k2$ for amplitude and phase are both set as 1.5. During training, the batch size is set as 300 with a learning rate of 0.0002. The complex weights are randomly initialised to a ball with small radius to achieve the bounded behaviour. For comparison, we also develop a real-valued neural network (RVNN) system under the same configuration except real-valued weights and inputs/outputs are applied. Some generated samples are available online for the reader to hear[2].

3.2. Experiment

To test the CVNN on real valued data, RDC features are first applied as the spectral representation. Both input and output in system INT-En-C (Table 1) are phase-encoded. The trajectory of the 2nd RDC for one utterance is shown in Fig. 1. We can see that the CVNN can predict a reasonable trajectory (red) compared with the natural one (blue). Then, we further apply this phase encoded system to the highly correlated log amplitude features (system DIR-En-C). The natural and generated trajectories of $logF_0$, vuv and 2nd log amplitude for one utterance is shown in Fig. 2 (left). We can see the CVNN can also generate reasonable trajectories for those features. For the real-valued data, we can also apply the traditional RVNN system to map the real-valued inputs to outputs directly. Therefore, here we also train an RVNN system (DIR-Ze-R) to map the real valued linguistic input to log amplitudes. Table 2 shows that while the same number of neurons, layers and activation function are applied, using the CVNN can result in smaller errors than the RVNN system.

[1] This learning rate parameter δ can be real, imaginary or complex valued

[2] http://homepages.inf.ed.ac.uk/s1164800/CVNN.html

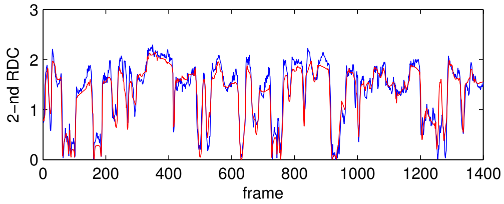

Fig. 1. Trajectories of predicted and natural 2-nd RDC for INT-En-C (blue: natural; red: generated)

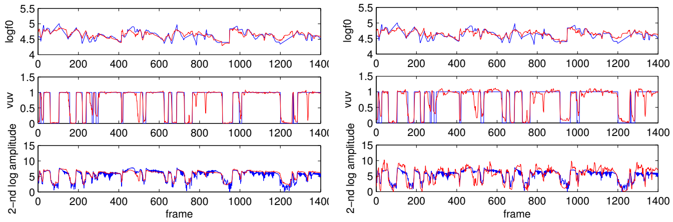

Fig. 2. Trajectories of predicted and natural logF0, vuv, 2nd log amplitude (left: DIR-En-C, right: CDIR-Mi-C; blue: natural, red: generated)

Finally, we test the ability of CVNNs for modelling complex valued acoustic features (system CDIR-Mi-C). In this paper, to avoid the influence of inaccurate disperse phase calculation from the sparse representation of sinusoids, only complex amplitudes with minimum phase extracted from a fixed number of sinusoids [25] are used as the spectral representation. For linguistic context features, $logF_0$ and vuv, phase coding is applied. From Fig. 3, we can see that error for both amplitude and phase decreases with epoch for training and testing data. The generated trajectories of vuv, $logF_0$, and the 2nd log amplitude are shown in Fig. 2 (right). Compared with result trained from DIR-En-C, CDIR-Min-C can also predict a similar trajectory for amplitude. Meanwhile, we also plot the minimum phase trajectory of the 2nd complex amplitude predicted from the CVNN (red) with the natural one (blue) in Fig. 4. We can see that it can also generate a reasonable phase trajectory.

Table 2. *Objective results for CVNN and RVNN systems*

ID	log amplitude	vuv	f0
	RMSE (dB)	rate (%)	RMSE (Hz)
DIR-Ze-RVNN	5.57	5.20	10.18
DIR-En-CVNN	5.44	3.44	10.17

4. CONCLUSION

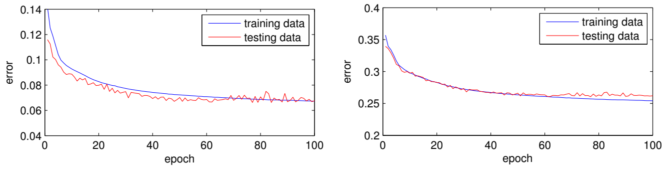

Fig. 3. RMSE for amplitude (left) and phase (right) for CDIR-Mi-C (blue: training data; red: testing data)

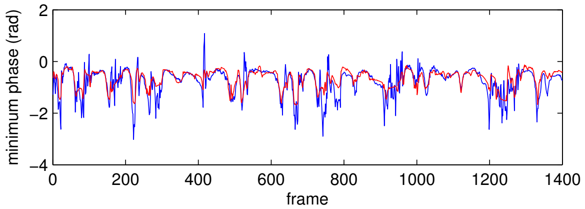

Fig. 4. Trajectories of the minimum phase for predicted and natural 2nd complex amplitude for CDIR-Mi-C (blue: natural; red: generated)

Complex valued analysis in the frequency domain is a method that is used often for speech signals. But most statistical models are designed for real-valued data. This paper mainly introduces a complex valued neural network for SPSS and investigates methods to model both real and complex valued acoustic signals using the proposed system. A fully complex valued feed-forward network is applied for speech synthesis with complex valued weights, activation function and learning algorithm. Real valued data is phase encoded prior to CVNN processing. Log amplitudes with minimum phase extracted from PDM is applied as complex valued output. Our results show the potential of using CVNNs for modelling both real and complex valued acoustic features.

However, when interpreting the experiment, it is necessary to bear in mind certain caveats. First, although results indicate the CVNN's ability to model both amplitude and phase, only minimum phase is modelled here. This is normally derived from generated amplitude and does not convey new information. So the performance of using other phase representations (e.g. disperse phase) still needs to be tested. Second, objective results show that for real-valued log amplitudes, CVNNs outperform the traditional RVNN, but the weights contained in the former system are complex, so their dimensionality is almost doubled compared to the RVNN. Finally, although the systems can generate speech with reasonable quality by using only two hidden layers with 100 nodes each, a listening test has not been conducted in this preliminary work, as the framework needs to be further refined to improve voice quality. Further work will focus on using CVNNs for modelling other type of phase representation with greater numbers of neurons and layers.

5. ACKNOWLEDGEMENTS

The authors greatly appreciate the help given by Dr. Zhizheng Wu for providing the bottleneck features and for suggestions given by Prof. Sundaram Suresh, Dr. Ramasamy Savitha and Dr. Ranniery Maia during the experiments.

6. REFERENCES

[1] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *Proc. 6th SSW*, 2007.

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

[3] V. Tsiaras, R. Maia, V. Diakoloukas, Y. Stylianou, and V. Digalakis, "Linear dynamical models in speech synthesis," in *Pro. ICASSP*, 2014.

[4] K. Paliwal, K. Wójcicki, and B. Shannon, "The importance of phase in speech enhancement," *speech communication*, vol. 53, no. 4, pp. 465–494, 2011.

[5] R. Schluter and H. Ney, "Using phase spectrum information for improved speech recognition performance," in *Pro. ICASSP*, 2001.

[6] R. Maia, M. Akamine, and M. Gales, "Complex cepstrum for statistical parametric speech synthesis," *Speech Communication*, vol. 55, no. 5, pp. 606–618, 2013.

[7] I. Saratxaga, I. Hernaez, D. Erro, E. Navas, and J. Sanchez, "Simple representation of signal phase for harmonic speech models," *Electronics letters*, vol. 45, no. 7, pp. 381–383, 2009.

[8] A. Stark and K. Paliwal, "Group-delay-deviation based spectral analysis of speech.," in *INTERSPEECH*, 2009, pp. 1083–1086.

[9] Y. Agiomyrgiannakis and Y. Stylianou, "Wrapped gaussian mixture models for modeling and high-rate quantization of phase data of speech," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 17, no. 4, pp. 775–786, 2009.

[10] G. Degottex, A. Roebel, and X. Rodet, "Function of phase-distortion for glottal model estimation," in *Pro. ICASSP*, 2011.

[11] G. Degottex and D. Erro, "A uniform phase representation for the harmonic model in speech synthesis applications," *EURASIP, Journal on Audio, Speech, and Music Processing - Special Issue: Models of Speech - In Search of Better Representations*, vol. 2014, no. 1, pp. 38, 2014.

[12] NR Goodman, "Statistical analysis based on a certain multivariate complex gaussian distribution (an introduction)," *Annals of mathematical statistics*, pp. 152–177, 1963.

[13] H. Leung and S. Haykin, "The complex backpropagation algorithm," *Signal Processing, IEEE Transactions on*, vol. 39, no. 9, pp. 2101–2104, 1991.

[14] R. Savitha, S. Suresh, N. Sundararajan, and P. Saratchandran, "A new learning algorithm with logarithmic performance index for complex-valued neural networks," *Neurocomputing*, vol. 72, no. 16, pp. 3771–3781, 2009.

[15] Aizenberg I, *Complex-valued neural networks with multi-valued neurons*, vol. 353, Springer, 2011.

[16] T. Kim and T. Adalı, "Approximation by fully complex multilayer perceptrons," *Neural Computation*, vol. 15, no. 7, pp. 1641–1666, 2003.

[17] A. Hirose, *Complex-valued neural networks: theories and applications*, vol. 5, World Scientific Publishing Company Incorporated, 2003.

[18] M. Amin and K. Murase, "Single-layered complex-valued neural network for real-valued classification problems," *Neurocomputing*, vol. 72, no. 4, pp. 945–955, 2009.

[19] Q. Hu, Y. Stylianou, K. Richmond, R. Maia, J. Yamagishi, and J. Latorre, "A fixed dimension and perceptually based dynamic sinusoidal model of speech," in *Proc. ICASSP*, 2014.

[20] T. Kim and T. Adali, "Fully complex multi-layer perceptron network for nonlinear signal processing," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 32, no. 1-2, pp. 29–43, 2002.

[21] S. Suresh, N. Sundararajan, and R. Savitha, *Supervised learning with complex-valued neural networks*, Springer, 2013.

[22] K. Richmond, P. Hoole, and S. King, "Announcing the electromagnetic articulography (day 1) subset of the mngu0 articulatory corpus.," in *Proc. Interspeech*, 2011.

[23] Z. Wu, C. Botinhao, O. Watts, and S. King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *Proc. ICASSP*, 2015.

[24] Q. Hu, Y. Stylianou, R. Maia, K. Richmond, J. Yamagishi, and J. Latorre, "An investigation of the application of dynamic sinusoidal models to statistical parametric speech synthesis," in *Proc. Interspeech*, 2014.

[25] Q. Hu, Y. Stylianou, R. Maia, K. Richmond, and J. Yamagishi, "Methods for applying dynamic sinusoidal models to statistical parametric speech synthesis," in *Proc. ICASSP*, 2015.

[26] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, 2000.