High-quality waveform generator from fundamental frequency, spectral envelope, and band aperiodicity

Masanori Morise*† and Takuro Shono‡
* School of Interdisciplinary Mathematical Sciences, Meiji University, Japan
E-mail: mmorise@meiji.ac.jp Tel/Fax: +81-3-5343-8332
† JST, PRESTO, Japan
‡ Integrated Graduate School of Medicine, Engineering, and Agricultural Sciences, University of Yamanashi, Japan

***Abstract*—This paper introduces a waveform generation algorithm from three speech parameters (fundamental frequency f_o, spectral envelope, and band aperiodicity). The conventional speech analysis/synthesis system based on a vocoder mainly has a waveform generator based on pitch synchronous overlap and add (PSOLA). Since it uses the fast Fourier transform (FFT) to generate the vocal cord vibration, the processing speed is proportional to the f_o. The algorithm also uses the spectral representation of the aperiodicity, whereas the band aperiodicity is mainly used in speech synthesis applications such as statistical parametric speech synthesis. We propose a waveform generation algorithm that reduces the computational cost and memory usage without degrading the synthesized speech. The algorithm utilizes excitation signal generation by directly using the band aperiodicity. The computational cost in a certain period is fixed because the excitation signal is filtered and processed by the overlap-add (OLA) algorithm. We used the re-synthesized speech to perform two evaluations for the processing speed and sound quality. The results showed that the sound quality of speech synthesized was almost the same by our proposed algorithm as by the conventional algorithm. The proposed algorithm can also reduce computational cost and memory usage.**

I. Introduction

Statistical parametric speech synthesis (SPSS) [1] is used in text-to-speech (TTS) synthesis, and the number of deep neural networks (DNNs) is increasing [2]. Many SPSS systems use a high-quality vocoder [3]. It decomposes the speech signal into the fundamental frequency f_o, spectral envelope, and aperiodicity. WaveNet [4] was proposed in 2016 as a TTS system that requires no vocoder, implying that a vocoder may not be required for TTS research in the future. On the other hand, a vocoder remains useful for applications such as voice morphing [5] and real-time voice conversion [6], [7].

Several high-quality vocoders have been proposed to achieve various kinds of applications. STRAIGHT [8] is one of the best due to its advanced signal processing algorithm, and we have proposed other solutions such as TANDEM-STRAIGHT [9], [10] and WORLD [11]. In particular, WORLD (D4C edition [12]) has been used for recent applications such as Merlin [13], a neural parametric singing synthesizer [14], and for voice synthesis using voices sampled from in-the-wild speakers [15]. Its waveform generation algorithm can synthesize natural speech but has a heavy computational cost and large memory usage. We seek to overcome these problems without degrading the synthesized speech. The current version of WORLD consists of several algorithms for estimating the speech parameters. In this paper, we use Harvest [16] for the f_o, CheapTrick [17], [18] for the spectral envelope, and D4C [12] for the aperiodicity estimators.

The rest of this paper is organized as follows. In Section 2, we discuss related works on waveform generation from speech parameters and outline our algorithm. In Section 3, we detail our algorithm. In Section 4, we compare our algorithm with the current WORLD regarding sound quality and processing speed of the re-synthesized speech through two evaluations. In Section 5, we conclude with a brief summary and mention of future work.

II. Related works and concept of proposed algorithm

Waveform generation algorithms from vocoder parameters have been proposed for various purposes. The timbre of the speech synthesized was "buzzy" in traditional algorithms [19], so a mixed-source model [20] and a multiband excitation vocoder [21] have been proposed. Aperiodicity finely controls the buzzy timbre, so STRAIGHT uses an accurate aperiodicity estimator [22]. Aperiodicity modeling by sinusoidal function [23] has been proposed for high-quality waveform generation. PLATINUM [24], Vocine [25], and a log domain pulse model [26], are also solutions proposed. A waveform generator based on neural network has also been proposed recently [27].

When processing, a vocoder-based algorithm requires an FFT to generate the excitation signal for approximating the vocal cord vibration. Therefore, the processing speed strongly depends on the f_o related to the number of vocal cord vibrations. To improve the processing speed, the number of FFTs per a certain period should be reduced independently of the f_o. This is one restriction of our algorithm.

A mel-log spectrum approximate (MLSA) filter [28] has been proposed for efficient memory usage. The MLSA filter generates the waveform from the mel-cepstrum obtained from the mel-cepstral analysis [29], [30]. Since the mel-cepstrum is a coded parameter, the memory usage can be reduced using the MLSA filter. On the other hand, the MLSA filter has been optimized for narrow band (16 kHz sampling) speech.

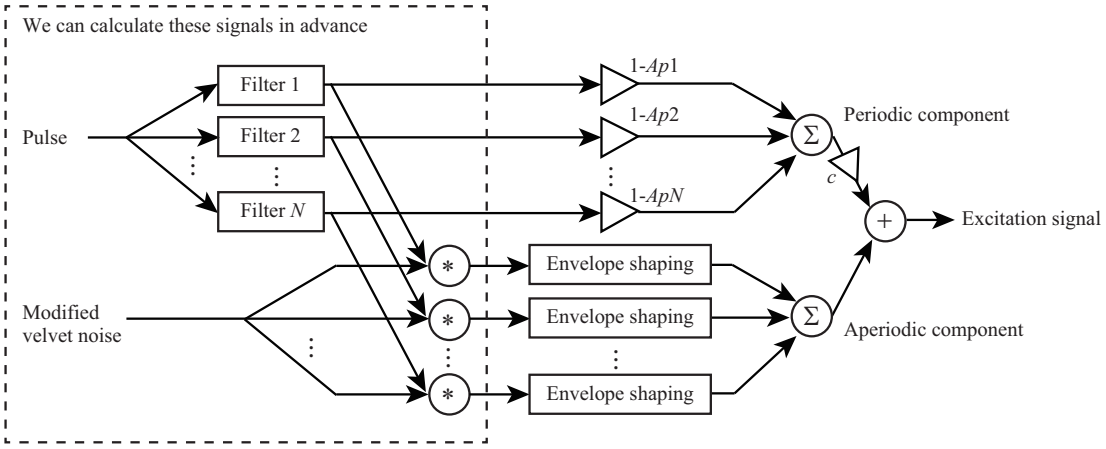

Fig. 1. Overview of the proposed excitation signal generation illustrating how to generate one vocal cord vibration, and how the processing speed for generating the whole signal depends on the f_{o}.

Parameter optimization is required for full-band speech with sampling frequencies above 40 kHz.

In this paper, we focus on the band aperiodicity. Band aperiodicity is used in mixed excitation linear prediction coding [31], and our previous work showed that the band aperiodicity sufficiently synthesizes natural speech [12]. We demonstrated that band aperiodicity consisting of five frequency bands could synthesize natural speech even if the input is full-band speech. However, the spectral representation of the aperiodicity is required with the conventional algorithm, and the required sample is the same as the spectral envelope. In cases where the FFT size is set to 2,048 (the default for WORLD), the memory usage in one frame is for 1,025 samples in both the spectral envelope and the aperiodicity.

To overcome the two previously mentioned points, we attempt to improve the processing speed and reduce the memory usage without degrading the synthesized speech. First, we directly use the band aperiodicity in excitation signal generation, and then we apply a simple overlap-add (OLA) by using the time-domain filter designed by the spectral envelope.

III. Algorithm details

Our algorithm consists of two steps. The first step is the excitation signal generation shown in Fig. 1. This figure shows the excitation signal design for one vocal cord vibration. The generated signal is processed on the basis of the pitch synchronous overlap and add (PSOLA) technique [32].

The generation shown in Fig. 1 includes the FFT, but we can calculate the filter and the convolution in advance. PSOLA-based processing depends on the f_{o}, but the dependency can be reduced because this processing that does not use FFT is negligibly low compared with FFT. The coefficient c in the figure is determined as the square root of the sample between two vocal cord vibrations.

A. Frequency band division

The excitation signal is designed in each frequency band, and the periodic and aperiodic components are independently generated as shown in Fig. 1. Band division is carried out using several band-pass filters. A quadrature mirror filter (QMF) [33] is typically used for general band division, but we employed a filter bank with a spectrum as shown in Fig. 2. The amplitude spectrum is calculated by a Hanning window with a width of 6 kHz. The center frequencies are the whole-number multiple of 3 kHz, which is the same as the center frequency of the band aperiodicity obtained from D4C. The duration of impulse response is reduced using such a smooth amplitude spectrum [34].

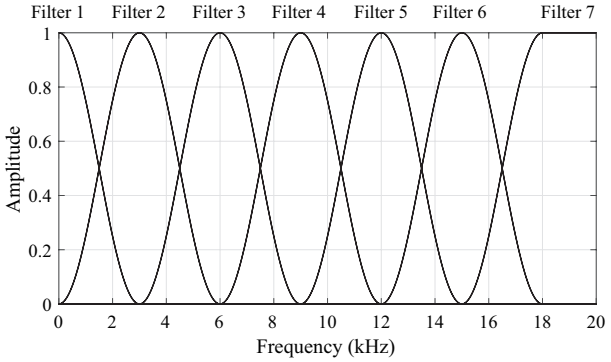

Fig. 2. Amplitude spectra of the designed filter bank. The summation of all spectra is completely flat.

The number of filters is determined by the sampling frequency set to 7 in full-band speech, but the band aperiodicity used in filters 1 and 7 is determined automatically [12]. In this case, the aperiodicity is -60 dB in filter 1 and 0 dB in filter 7, allowing us to control the values of five frequency bands. Zero phase is used in all filters to reconstruct the input signal completely. An acausal component is generated, but the duration of the impulse response is sufficiently short from the aspect of auditory masking.

B. Noise generation used for aperiodic component

Conventional algorithms used the white noise to generate the aperiodic component. The comprehensive power spectrum is flat, but short-term white noise often generates the component at 0 Hz. This component is removed before the

convolution because it causes noise in the synthesized speech. We employ a new noise called "modified velvet noise" (MVN) [35] instead of white noise. Here, we explain how to generate velvet noise [36], [37] because MVN is a modified version.

First, we determine the impulse locations with the following equation.

$$k_{\mathrm{ovn}}(m) = ||mT_{\mathrm{d}} + r_1(m)(T_{\mathrm{d}} - 1)||, \quad (1)$$

where m represents the pulse counter ($m = 0, 1, 2, ...$), $||x||$ represents the rounding function for the input x, $r_1(m)$ represents a sequence of random numbers uniformly distributed in the range from 0.0 to 1.0, and T_d represents the average distance of impulses. The velvet noise $s_{\mathrm{ovn}}(n)$ is given by

$$s_{\mathrm{ovn}}(n) = \begin{cases} 2||r_2(m))|| - 1, & n = k_{\mathrm{ovn}}(m) \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$r_2(m)$ represents another sequence of random numbers. The amplitude of velvet noise consists of 1 and -1. Velvet noise has almost the same power spectrum as that of white noise.

To generate MVN, we define the sample N as the basic length of $s_{\mathrm{ovn}}(n)$, and the parameter T_d is set to 4. In cases where N is set to a multiple of 8, the number of pulses is fixed to an even number. We unify the number of 1 and -1 to the same number by controlling the random number. This modification guarantees that there is no component at 0 Hz. When a long length above N is required, N sample velvet noises generated with different random seeds are concatenated and extracted by the required sample.

The component at 0 Hz can be removed by using short-term N. However, velvet noise generated with this approach peaks at a frequency f_s/N Hz, where f_s represents the sampling frequency. To overcome this problem, MVN uses two short-term velvet noises generated with the different samples (N_1 and N_2) randomly selected and concatenated to reduce the frequency peak.

Fig. 3 illustrates the power spectrum of an MVN. The power spectrum was calculated 10,000 times by using different random seeds, and the results are their averages. The signal sample was set to 8,192 samples, and the sampling frequency was set to 48 kHz. Samples N_1 and N_2 were set to 400- and 152-sample shown in the paper [35]. These samples obtain an approximately flat power spectrum ($\pm$0.6 dB) from 100 Hz to the Nyquist frequency (24 kHz). As expected, low frequency noise can be reduced using this noise. In cases where the amplitude is set to 2 and -2, the power of the noise is the same as that of the white noise.

C. Envelope shaping

After speech analysis, a band aperiodicity $A_p(n)$ is obtained in each frequency band. This is the discrete time sequence, and n represents the discrete time. Since the aperiodicity $A_p(n)$ has the value at the time as a whole-number multiple of the frame shift, the aperiodicity for shaping is calculated by a simple linear interpolation. The temporal envelope of the MVN in each frequency band is shaped by using the interpolated

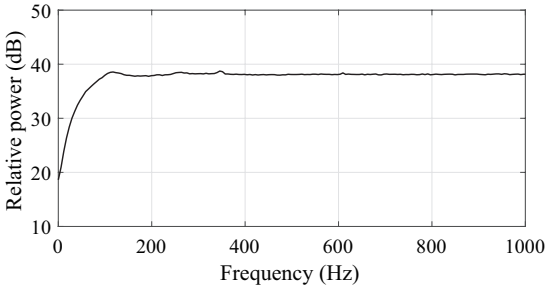

Fig. 3. Power spectrum of the MVN (400- and 152-sample). These samples obtain an approximately flat power spectrum ($\pm$0.6 dB) from 100 Hz to the Nyquist frequency (24 kHz).

aperiodicity. This process enables us to generate a noise that has a temporally smooth change.

D. Overlap-add process

A simple OLA is carried out after generating the excitation signal. The impulse response of the filter is designed using the spectral envelope. Since the minimum phase is considered better than the zero phase [38], our algorithm uses the minimum phase to calculate the impulse response. This process depends on the frame shift instead of the f_{o}.

E. Advantages of proposed algorithm

The conventional algorithm requires an independent process for synthesizing the periodic and aperiodic components. Since our algorithm can generate both components by using one filter, the number of FFTs is reduced to half. In the conventional algorithm, the number of vocal cord vibrations determines the number of FFTs. Our algorithm requires the fixed times to be based on the frame shift.

Our algorithm can directly use the band aperiodicity, thereby using less memory than the conventional algorithm does. The total number of dimensions of the three parameters per frame is 2,051 (f_{o}: 1, spectral envelope: 1,025, and aperiodicity: 1,025) in the conventional algorithm. On the other hand, only 1,031 dimensions are required (f_{o}: 1, spectral envelope: 1,025, and aperiodicity: 5) in our algorithm. Our algorithm uses about half the memory the conventional algorithm does.

IV. Evaluation and discussion

We carried out two evaluations to demonstrate the effectiveness of our algorithm. One was an AB preference test using original and re-synthesized speech. The other was an objective evaluation to compare the processing speed of our algorithm with the conventional algorithm's processing speed. Several algorithms for waveform generation have been proposed, but we only used the conventional algorithm used in WORLD for comparison. We evaluated sound quality for comparison [39], and the results showed that WORLD was significantly better than STRAIGHT.

Since the processing speed depends on the implementation, fair evaluation of our algorithm compared with those implemented by other developers was difficult. STRAIGHT

TABLE I
EXPERIMENTAL CONDITIONS.

Evaluation protocol	
Method	AB preference test
Number of subjects	12 persons
Environment	Soundproof room
Background noise	18 dB (A-weighted SPL)
Headphones	SENNHEISER HD650
Audio I/O	Roland QUAD-CAPTURE
Characteristics of the speech used in the evaluation	
Number of speakers	4 (2 men and 2 women)
Number of stimuli	20 (5 words per speaker)
Kind of speech	4-mora word including consonants
Sampling	48 kHz/16 bit

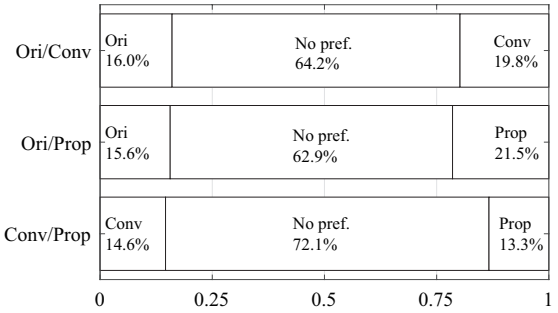

Fig. 4. Result of the AB preference test. Both algorithms can synthesize speech as naturally as the input waveform.

and TANDEM-STRAIGHT use similar algorithms, and this evaluation would provide us with an indirect comparison. In our algorithm, we set the frame shift used in OLA to 5 ms, the default frame shift for the analysis. The FFT size used for generating the excitation signal was set to 1,024 (around 21.3 ms).

A. Subjective evaluation by using original and re-synthesized speech

Table I shows the conditions of the subjective evaluation. Twelve subjects with normal hearing ability participated in the evaluation. The subjects were asked to listen to two stimuli and tell their preference. In cases where they could not identify the difference, they answered "no preference." The speech stimuli used in the evaluation were selected based on the re-synthesized speech's sound quality. To verify the difference between algorithms, we manually selected the stimuli that included no estimation errors.

Fig. 4 illustrates the results of the evaluation. In this figure, Ori represents original speech, Conv represents the conventional algorithm, and Prop represents the proposed algorithm. In all combinations, speech stimuli of at least 62.9% were judged as no preference ("No pref." in the figure). The results showed that WORLD-based algorithms can synthesize speech just as naturally as original speech. The rate of no preference was 72.1% between the conventional algorithm and our algorithm, showing that the sound quality of our algorithm was almost the same as that of the conventional algorithm.

B. Processing speed evaluation by real-time factor

We evaluated two algorithms in terms of the real-time factor (RTF). We used a mobile PC (Intel Core i7-7500U CPU 2.70 GHz and 16.0 GB RAM) in the evaluation. The algorithms were implemented on MATLAB (Version R2016b). No parallel processing was used in the evaluation.

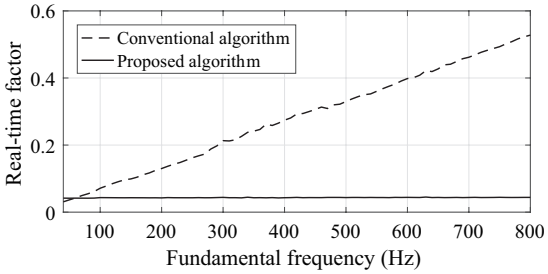

Fig. 5. Results of the RTF evaluation. The processing speed linearly increased in the conventional algorithm. However, the proposed algorithm can remove this dependency.

The evaluation was to verify the relationship between the f_{o} and the processing speed. The target signal was a pulse train with various kinds of f_{o}, and speech parameters were artificially given. The band aperiodicity was set to -3 dB in all frequency bands. The duration of the signal was set to 1 s to calculate the RTF. The RTF was calculated 100 times in each f_{o}, and its median value was used as the result. We used the frequency range 40 to 800 Hz in the evaluation.

Fig. 5 illustrates the results. The horizontal axis represents the f_{o}, and the vertical axis represents the RTF. The conventional algorithm strongly depends on the f_{o}, but our algorithm removed that dependency. The average RTF of our algorithm was around 0.045, implying that our algorithm is effective for real-time applications.

C. Discussion

The results showed that our algorithm can work as expected. The sound quality of the re-synthesized speech was almost the same as that of the original speech. Our algorithm was superior to the conventional one in both processing speed and memory usage. The f_{o} dependency in the processing speed could be solved completely in the frequency range (40-800 Hz). We can reduce memory usage further by using the MLSA filter with the parameter tuning for full-band speech. Comparisons with other high-quality systems based on other concepts, such as a sinusoidal model [40] or a phase vocoder, is also important.

In our algorithm, 15.6% of the re-synthesized speech could not be synthesized to the same quality. In particular, the results of the male speech showed a clear difference (15.0% vs. 10.0%), whereas the difference was relatively small in the female speech (14.2% vs. 16.7%). This result suggests that the sound quality depends on the f_{o}. A human being is typically sensitive to the phase difference in low f_{o} speech. If the conventional algorithm controls the fractional delay below 1 sample in the vocal cord vibration, the difference would be caused by this difference. We will perform a more extensive evaluation to examine the cause of this result.

V. CONCLUSION

We proposed a waveform generation algorithm based on excitation signal generation. The memory usage of our algo-

rithm was around half that of a conventional algorithm. The processing speed was better than that of the conventional one, and the dependency on f_o was also removed. Furthermore, our algorithm could achieve a sound quality of speech almost the same as that of the conventional algorithm and the original waveform.

The next step of this research is a significant evaluation using the re-synthesized speech and converted speech. Since the parameters used in the experiment were determined by only the exploratory evaluation, parameter optimization using the above evaluation is important. Implementation of a real-time processing application such as real-time STRAIGHT [6] is also important for many applications.

VI. Acknowledgment

This work was supported by JSPS KAKENHI Grant Numbers JP16H01734 and JP16H05899, and JST PRESTO Grant Number JPMJPR18J8, Japan.

References

[1] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, pp. 1039–1064, 2009.

[2] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," *in Proc. ICASSP2013*, pp. 7962–7966, 2013.

[3] H. Dudley, "Remaking speech," *J. Acoust. Soc. Am.*, vol. 11, no. 2, pp. 169–177, 1939.

[4] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[5] H. Kawahara, R. Nisimura, T. Irino, M. Morise, T. Takahashi, and H. Banno, "Temporally variable multi-aspect auditory morphing enabling extrapolation without objective and perceptual breakdown," *in Proc. ICASSP2009*, pp. 3905–3908, 2009.

[6] H. Banno, H. Hata, M. Morise, T. Takahashi, T. Irino, and H. Kawahara, "Implementation of realtime straight speech manipulation system," *Acoust. Science & Technology*, vol. 28, no. 3, pp. 140–146, 2007.

[7] M. Morise, M. Onishi, H. Kawahara, and H. Katayose, "v.morish'09: A morphing-based singing design interface for vocal melodies," *Lecture Notes in Computer Science*, vol. LNCS 5709, pp. 185–190, 2009.

[8] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive timefrequency smoothing and an instantaneous-frequency-based F0 extraction," *Speech Communication*, vol. 27, no. 3–4, pp. 187–207, 1999.

[9] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno, "TANDEM-STRAIGHT: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, f0, and aperiodicity estimation," *in Proc. ICASSP2008*, pp. 3933–3936, 2008.

[10] H. Kawahara and M. Morise, "Technical foundations of TANDEM-STRAIGHT, a speech analysis, modification and synthesis framework," *SADHANA - Academy Proceedings in Engineering Sciences*, vol. 36, no. 5, pp. 713–728, 2011.

[11] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Trans. Inf. & Syst.*, vol. E99-D, pp. 1877–1884, 2016.

[12] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Communication*, vol. 84, pp. 57–65, 2016.

[13] Z. Wu, O. Watts, and S. King, "Merlin: An open source neural network speech synthesis system," *in Proc. SSW 2016*, pp. 218–223, 2016.

[14] M. Blaauw and J. Bonada, "A neural parametric singing synthesizer modeling timbre and expression from natural songs," *Applied Science*, vol. 7, no. 12, pp. 23–page, 2018.

[15] Y. Taigman, L. Wolf, A. Polyak, and E. Nachmani, "VoiceLoop: Voice fitting and synthesis via a phonological loop," *in Proc. ICLR 2018*, pp. 14–page, 2018.

[16] M. Morise, "Harvest: A high-performance fundamental frequency estimator from speech signals," *in Proc. INTERSPEECH2017*, pp. 2321–2325, 2017.

[17] ——, "CheapTrick, a spectral envelope estimator for high-quality speech synthesis," *Speech Communication*, vol. 67, pp. 1–7, 2015.

[18] ——, "Error evaluation of an f0-adaptive spectral envelope estimator in robustness against the additive noise and f0 error," *IEICE Trans. Inf. & Syst.*, vol. E98-D, no. 7, pp. 1405–1408, 2015.

[19] O. Fujimura, "An approximation to voice aperiodicity," *IEEE Trans. on Audio and Electroacoust.*, vol. 16, no. 1, pp. 68–72, 1968.

[20] J. Makhoul, R. Viswanathan, R. Schwartz, and A. Huggins, "A mixed-source model for speech compression and synthesis," *J. Acoust. Soc. Am.,*, vol. 64, no. 6, pp. 1577–1581, 1978.

[21] D. W. Griffin and J. S. Lim, "Multiband excitation vocoder," *IEEE Trans. on Speech, and Signal Process.*, vol. 36, no. 8, pp. 1223–1235, 1988.

[22] H. Kawahara, J. Estill, and O. Fujimura, "Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and synthesis system STRAIGHT," *in Proc. MAVEBA2001*, pp. 59–64, 2001.

[23] H. Kawahara, M. Morise, T. Takahashi, H. Banno, R. Nisimura, and T. Irino, "Simplification and extension of non-periodic excitation source representations for high-quality spee3ch manipulation systems," *in Proc. INTERSPEECH2010*, pp. 38–41, 2010.

[24] M. Morise, "PLATINUM: A method to extract excitation signals for voice synthesis system," *Acoust. Sci. & Tech.*, vol. 33, no. 2, pp. 123–125, 2012.

[25] Y. Agiomyrgiannakis, "Vocaine the vocoder and applications in speech synthesis," *in Proc. ICASSP 2015*, pp. 4230–4234, 2015.

[26] G. Degottex, P. Lanchantin, and M. Gales, "A log domain pulse model for parametric speech synthesis," *IEEE/ACM Trans. on audio, speech, and language process.*, vol. 26, no. 1, pp. 57–70, 2018.

[27] K. Kobayashi, T. Hayashi, A. Tamamori, and T. Toda, "Statistical voice conversion with WaveNet-based waveform generation," *in Proc. INTERSPEECH 2017*, pp. 1138–1142, 2017.

[28] S. Imai, K. Sumita, and C. Furuichi, "Mel log spectrum approximation (MLSA) filter for speech synthesis," *Electronics and Communications in Japan (Part I Communications)*, vol. 66, no. 2, pp. 10–18, 1983.

[29] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," *in Proc. ICASSP'92*, vol. 1, pp. 137–140, 1992.

[30] K. Tokuda, "Speech coding based on adaptive melcepstral analysis," *in Proc. ICASSP'94*, pp. 197–200, 1994.

[31] W. Lin, S. N. Koh, and X. Lin, "Mixed excitation linear prediction coding of wideband speech at 8 kbps," *in Proc. ICASSP'00*, vol. 2, pp. 1137–1140, 2000.

[32] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Communication*, vol. 9, no. 5-6, pp. 453–467, 1990.

[33] J. Johnston, "A filter family designed for use in quadrature mirror filter banks," *in Proc. ICASSP'80*, pp. 291–294, 1980.

[34] L. Cohen, *Time-frequency analysis*. Prentice Hall, 1994.

[35] M. Morise, "Modification of velvet noise for speech waveform generation by using vocoder-based speech synthesizer," *IEICE Trans. Inf. & Syst.*, vol. E102-D, no. 3, pp. 663–665, 2019.

[36] M. Karjalainen and H. Järveläinen, "Reverberation modeling using velvet noise," *in Proc. AES 30th International conference*, pp. 9–page, 2007.

[37] V. Välimäki, H.-M. Lehtonen, and M. Takanen, "A perceptual study on velvet noise and its variants at different pulse densities," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 21, no. 7, pp. 1481–1488, 2013.

[38] A. V. Oppenheim, "Speech analysis-synthesis system based on homomorphic filtering," *J. Acoust. Soc. Am.*, vol. 45, no. 2, pp. 458–465, 1969.

[39] M. Morise and Y. Watanabe, "Sound quality comparison among high-quality vocoders by using re-synthesized speech," *Acoust. Sci. & Tech.*, vol. 39, no. 3, pp. 263–265, 2018.

[40] R. J. McAulay and T. F. Quatieri, "Speech processing based on a sinusoidal model," *The Lincoln Laboratory Journal*, vol. 1, no. 2, pp. 153–168, 1988.

[41] J. L. Flanagan and R. M. Golden, "Phase vocoder," *The Bell System Technical Journal*, vol. 45, no. 9, pp. 1493–1509, 2009.