Singing Voice Synthesis Using Differentiable LPC and Glottal-Flow-Inspired Wavetables

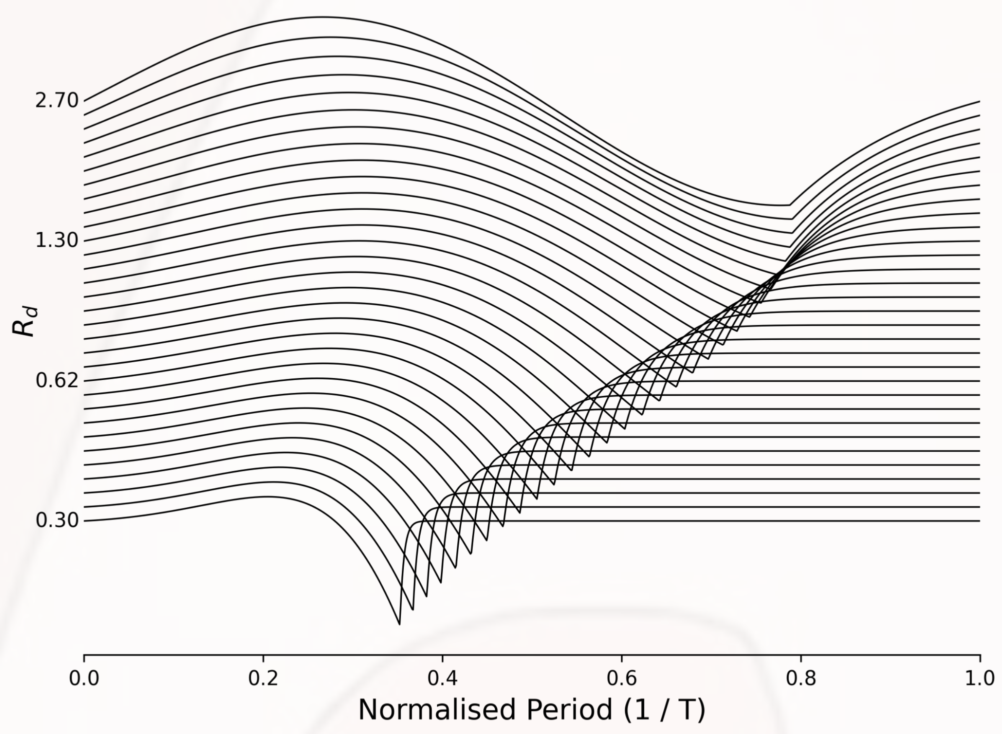
Chin-Yun Yu   György Fazekas
Centre for Digital Music, Queen Mary University of London

Motivations

The available DDSP-based singing voice synthesisers were not designed specifically for human voice. Leveraging well-examined voice production model could lead to a more efficient and interpretable neural voice synthesiser.

The Harmonics: Glottal Flow

- We sampled glottal flows from the Transformed-LF model [1] with R_d range from 0.3 to 2.7.
- The periodic signal is generated by wavetable synthesis, with time-varying fundamental frequency and R_d predicted by the encoder.



The filters: Differentiable LPC

Linear Predictive Coding (LPC)

$$s_n = e_n - \sum_{i=1}^{M} \alpha_i s_{n-i}$$

- It has been used to approximate **vocal tract** response for decades.
- Differentiable recursion is **slow** in deep learning frameworks.
- Evaluating LPC filter in the frequency domain = FIRs approximation.

Solution

- Writing custom backward functions
- Decomposing the backpropation into **2 LPC filtering (1 and 3) and one matrix multiplication (2)**

$$\frac{\partial s_n}{\partial a_i} = -s_{n-i} - \sum_{k=1}^{M} a_k \frac{\partial s_{n-k}}{\partial a_i} \quad (1)$$

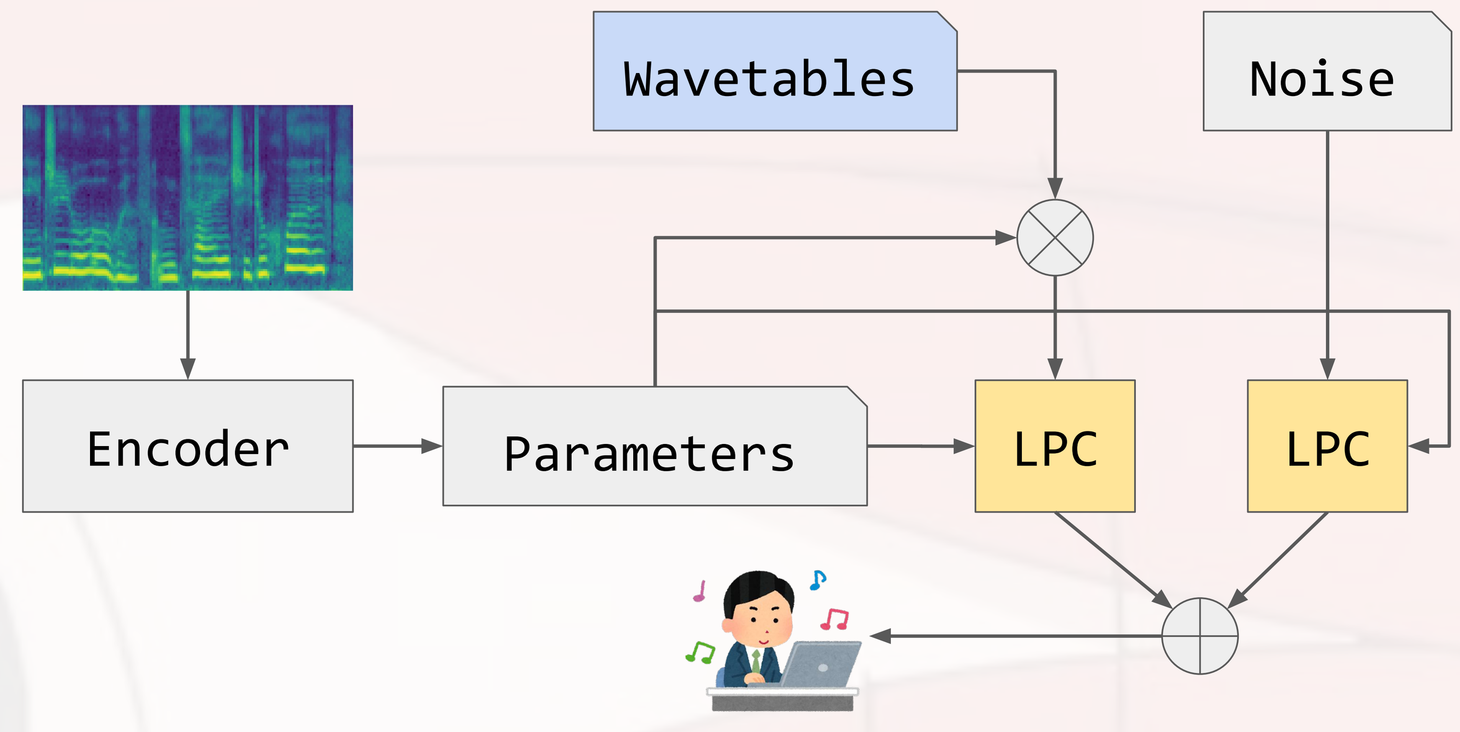
$$\frac{\partial \mathcal{L}}{\partial a_i} = \sum_{n=1}^{N} \frac{\partial \mathcal{L}}{\partial s_n} \frac{\partial s_n}{\partial a_i} \quad (2) \qquad \frac{\partial \mathcal{L}}{\partial e_n} = \frac{\partial \mathcal{L}}{\partial s_n} - \sum_{i=1}^{M} a_i \frac{\partial \mathcal{L}}{\partial e_{n+i}} \quad (3)$$

Pseudo Code in PyTorch

```
class DifferentiableLPC(torch.autograd.Function):
    @staticmethod
    def forward(ctx, e, alpha):
        s = fast_lpc(e, alpha)
        ctx.save_for_backward(s, alpha)
        return s

    @staticmethod
    def backward(ctx, grad_s):
        s, alpha = ctx.saved_tensors
        T, order = s.numel(), alpha.numel()
        # coefficient gradients
    (1) dsda = fast_lpc(-F.pad(s, (order, 0)), alpha).unfold(0, T, 1)
    (2) grad_alpha = dsda @ grad_s
        # input gradients
    (3) grad_e = fast_lpc(grad_s.flip(0), alpha).flip(0)
        return grad_e, grad_alpha
```

GlOttal-flow LPC Filter (GOLF) Vocoder



- **Frame-wise** LPC for time-varying synthesis
- Predicting **voiced/unvoiced** flag to eliminate harmonics in unvoiced sound (a.k.a the "buzzy" effect)

Experiments

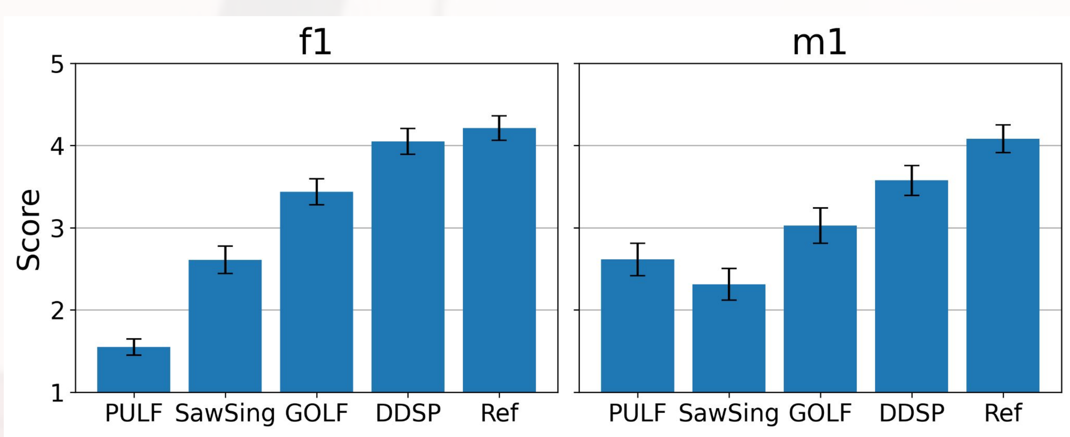
- Data: f1/m1 from MPop600 [2]
- Synthesisers: DDSP [3], SawSing [4], GOLF (ours), PULF (GOLF with pulse trains)
- Encoder: two CNN layers + three Bi-LSTM layers with 96 channels

Singers	Models	MSSTFT	MAE-f0 (cent)	FAD
f1	DDSP	**3.09**	**74.47**±1.19	0.50±0.02
	SawSing	3.12	78.91±1.18	**0.38**±0.02
	GOLF	3.21	77.06±0.88	0.62±0.02
	PULF	3.27	76.90±1.11	0.75±0.04
m1	DDSP	**3.12**	**52.95**±1.03	0.57±0.02
	SawSing	3.13	56.46±1.04	**0.48**±0.02
	GOLF	3.26	54.09±0.30	0.67±0.01
	PULF	3.35	54.60±0.73	1.11±0.04

GOLF and PULF are comparable to DDSP and SawSing regarding the mean absolute error (MAE) in F0.

Wavetables let GOLF require less memory to train and run **10 times faster** than baselines on CPU. Its waveform is also the most similar to the ground truth.

Models	Memory	RTF		Waveform L2	
		GPU	CPU	Min	Max
DDSP	7.3	0.015	0.237	71.83	88.77
SawSing	7.3	0.015	0.240	75.72	93.16
GOLF	**2.6**	**0.009**	**0.023**	**21.98**	**64.82**
PULF	7.5	0.015	0.248	44.08	70.59



GOLF surpasses SawSing significantly in subjective evaluation. The inferior result of PULF shows the importance of the glottal flow model.

Conclusions

- We proposed an efficient differentiable synthesiser based on the voice production model for neural voice synthesis.
- The low reconstruction errors on waveforms are a positive effect of using the glottal flow model and LPC filter, stating the importance of good inductive bias.
- Our filter implementation can be used directly in other tasks with recursive filters (i.e. LPC/IIR/All-Pole).

Reference

[1] G. Fant, "The LF-model revisited. transformations and frequency domain analysis," Speech Trans. Lab. Q. Rep., Royal Inst. of Tech. Stockholm, vol. 2, no. 3, p. 40, 1995.
[2] C.-C. Chu, F.-R. Yang, Y.-J. Lee, Y.-W. Liu, and S.-H. Wu, "MPop600: A mandarin popular song database with aligned audio, lyrics, and musical scores for singing voice synthesis," in APSIPA ASC. IEEE, 2020, pp. 1647– 1652.
[3] J. Engel, L. H. Hantrakul, C. Gu, and A. Roberts, "DDSP: Differentiable digital signal processing," in ICLR, 2020.
[4] D.-Y. Wu, W.-Y. Hsiao, F.-R. Yang, O. Friedman, W. Jackson, S. Bruzenak, Y.-W. Liu, and Y.-H. Yang, "DDSP-based singing vocoders: A new subtractivebased synthesizer and a comprehensive evaluation," in Proc. ISMIR, 2022.

UK Research and Innovation

Audio Samples

Code

Contact
Email: chin-yun.yu@qmul.ac.uk
GitHub/X: @yoyololicon