Project Report: Vowel Singing Synthesis with Anti-resonances

Xuehua Fu
KTH Royal Institute of Technology
xuehua@kth.se

Supervisor: Sten Ternström
KTH Royal Institute of Technology
stern@kth.se

ABSTRACT

The project implemented an advanced singing synthesizer that uses a time-reversed glottal flow model for the voice source and models the spectral notches in addition to formants to create a natural-sounding voice synthesis. The synthesizer was tuned to match several real singing voice examples. As a result, voice syntheses with both spectral notches and formants have better performance than those with only formants.

1. BACKGROUND

The project is originally inspired by the KTH synthesis of singing [1] project conducted by Johan Sundberg, Sten Ternström et. al.

The *Madde* synthesizer from *Tolvan Data* [1] has been a success in teaching voice science and pedagogy. It was based on a traditional source-filter approach to model the vocal folds source and vocal tract resonances. This project uses a similar approach to the Madde synthesizer and improves the vocal tract modeling.

2. METHOD

This project uses the traditional source-filter approach for modeling of the singing voice of human. The source-filter model describes human voice production as a two-stage process. Source is the sound produced by vocal folds. The source sound then travels through the vocal tract, which acts like a series of filters applied on the source. The filtering process results in a series of resonances of the voice source, which are decisive for the *vowel quality* [2] and *voice quality* [3] of the voice [2].

2.1 Voice source

For vocal folds phonation, the synthesis uses a time-reversed glottal flow model. The original oscillation is generated by the Band-Limited Impulse (Blip) oscillator with 200 harmonics by default. Then a resonant low-pass filter is applied to generate the time-reversed flow. A +12 dB/octave radiation characteristic is added for output as sound.

2.1.1 Vibrato and flutter

The vibrato in a real singing voice corresponds to a slow and nearly sinusoidal fundamental frequency modulation. To add a natural vibrato to the synthesis, the sound is multiplied by a sinusoidal oscillator with frequency at around 6.0 Hz and amplitude at around 0.015.

The flutter refers to a type random variations in a singing voice. It can be modeled by a white noise through a resonant low-pass filter centered at 4 to 6 Hz [3]. The flutter adds a natural variation to the synthesis in addition to the sinusoidal vibrato, which was also implemented in the Madde synthesizer.

2.2 Vocal-tract resonances and anti-resonances

The vocal-tract resonances manifest as a series of high spectrum envelope peaks of the singing voice, which are named *formants*. For real voices, formants correspond physically to shaping of the vocal tract, in other words, the *articulation*, in terms of jaw opening, tongue position, lip rounding, and the vocal tract length. The first two formants (which is going to be introduced again in Sec 3.1), corresponding to jaw opening and tongue positioning, determines the vowel quality of the voice.

Conventional methods use a series of second-order low-pass resonant filters applied on the source sound to obtain the spectral formants. State-Variable Filters (SVFs) [4] are used in this project as they provide three different outputs for low-pass, band-pass, and high-pass filtering. A mixed type of output is also allowed, referring to a linear combination of low-pass, band-pass, and high-pass outputs. The low-pass SVFs take the place of the second-order resonance filters to model vocal-tract formants in our synthesis.

With the low-pass filters, a rather smooth transition between the formant peaks is resulted. However, zero cavities are observed in the spectra of real singing voices, between or inside formants. These zero cavities are given by anti-resonances from the vocal tract.

A notch filter that captures the anti-resonance can be implemented using the SVF. The output of a notch filter is equivalent to the superposition of the high-pass output and the low-pass output of a SVF. Therefore, the spectral notches are modelled in the synthesis by using the notch filters.

In our synthesis, the formant filters and notch filters are both implemented with SVFs. The filters are applied in a serial manner, with centre frequencies up to 22050 kHz.

[1] Tolvan Data - Svante Granqvis. http://www.tolvan.com/

[2] *Vowel quality* refers to the type of vowel perceived.

[3] *Voice quality* is a personal characteristic that determines which individual produces the voice.

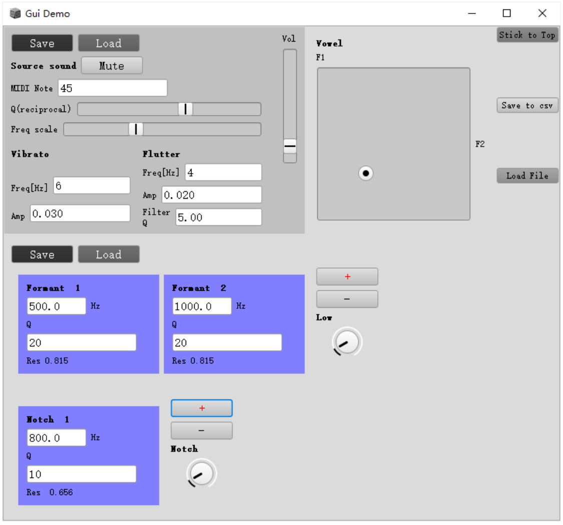
[4] Digital State-Variable Filters. https://ccrma.stanford.edu/~jos/svf/svf.pdf.

.

3. IMPLEMENTATION

The synthesizer is implemented in *SuperCollider-3.12.0* with a graphical control panel (Fig. 1) as the main user interface. A MIDI controller is programmed to be an alternative user interface of the synthesizer. The source code is available on GitHub [5] .

3.1 The synthesizer GUI



Source sound

MIDI Note 45

Q(reciprocal)

Freq scale

Vibrato

Freq[Hz] 6

Amp 0.030

Flutter

Freq[Hz] 4

Amp 0.020

Filter Q 5.00

Formant 1

500.0 Hz

Q 20

Res 0.815

Formant 2

1000.0 Hz

Q 20

Res 0.815

Notch 1

800.0 Hz

Q 10

Res 0.656

Figure 1: SuperCollider GUI of the synthesis.

There are mainly three modules on the graphical control panel: the source, the vowel control, and the two filter series.

The MIDI note refers the pitch of the voice source, namely the fundamental frequency of the voice. The Q factor and frequency scale are parameters of the synthesis glottal flow. Q controls the Q factor of the resonant low-pass filter that creates the flow, and frequency scale ranging from 1.2 to 3.0 scales the center frequency of the low-pass filter. Vibrato and flutter add a natural variation to the voice. Filter Q refers to the Q factor of the low-pass filter for the white noise to create a flutter [6] .

The vowel controller is a 2D slider for F1 and F2 frequencies, with F1 ranging from 200 Hz to 1200 Hz, and F2 ranging from 400 Hz to 2500 Hz. The first two formants of human singing voice, namely F1 and F2, indicate the jaw opening and tongue position respectively, and hence directly correspond to the vowel quality perceived. Figure 2 shows approximately how English vowels u, a, i, and ædistribute with respect to F1 and F2 in a 2D plot.

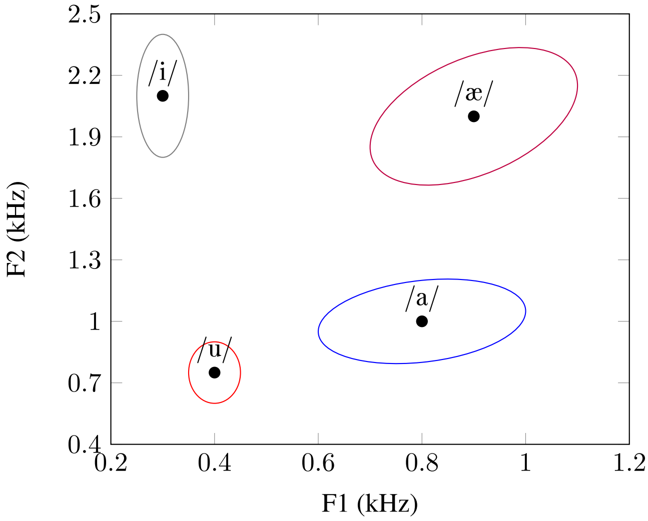


Figure 2: F1 and F2 for English vowels.

The two filter series include a series of low-pass formant filters, and a series of notch filters, both of which are derived from SVFs. The knob controls the output level of all low-pass filters or notch filters, ranging from 0 to 1. The $+/-$ buttons allow users to add a new formant/notch to the signal or remove a existing formant/notch. For each filter, the user is allowed to change the center frequency and the quality factor Q [7] . *Res* (0 to 1) is the *resonance* parameter of the SVF in the SuperCollider implementation [8] , and ”Q” is given by $res = 1 - (\frac{1}{Q})^4$ according to the tutorial of Digital State-Variable Filters from *CCRMA* [9] and the settings in the source code the SuperCollider implementation of the SVF. The center frequencies should be below 22050 kHz or half of the sample rate.

3.2 MIDI control

The synthesizer allows an external MIDI control device [10] . The keys controls the fundamental frequency of the voice. Eight sliders combined with eight knobs are for the formant and notch filters, which control the frequencies and Q factors respectively.

4. EVALUATION

Comparisons are made between the recordings of real singing voice and voice synthesis that simulate the corresponding singing voice. Each synthesis is tuned manually by analysing the spectrum of each real singing voice and matching the formant and notch frequencies by hand. The settings cover the source sound, formant filters and notch filters, and can be saved as a .csv file using the GUI (Table 1).

[5] The vowel singing synthesizer. `https://github.com/cheriestoner/Vowel-singing-synthesizer`.

[6] Yet the GUI contains flutter control, the implementation of the flutter is still problematic and is currently not involved in the actual synthesis.

[7] The Q factor usually refers to the proportion of center frequency to bandwidth of the filter. In the implementation of SVF used in this project, the Q factor is not explicitly defined, and instead, the parameter *res* is introduced, which turns out to be a function of the Q factor.

[8] SC3-Plugins source code. `https://github.com/supercollider/sc3-plugins/blob/main/source/BlackrainUGens`

[9] Digital State-Variable Filters - Julius Orion Smith III. `https://ccrma.stanford.edu/~jos/svf/svf.pdf`

[10] The synthesizer currently uses a KORG Kontrol MIDI keyboard from the KTH Speech, Music, and Hearing (TMH) department.

The .csv file can be reloaded to the synthesizer for further manipulation (See Fig. 1).

Pitch in MIDI note	50	Volume	0.742188							
1/Q	0.6									
Freq scale	0.510417									
Vibrato freq	4									
Vibrato amp	0.01									
Low level	0.19									
Formants										
658	1102	2800	3830	5736	6800	7500	8600	10246	12158	14072
Res										
0.814506	0.814506	0.903688	0.903688	0.814506	0.814506	0.814506	0.814506	0.814506	0.873186	0.873186
Q										
20	20	40	40	20	20	20	20	20	30	30
Notch level	0.15									
Notches										
850	1430	2220	4900	6600	9990					
Res										
0.6561	0.6561	0.6561	0.6561	0.6561	0.6561					
Q										
10	10	10	10	10	10					

Table 1: An example showing the format of synthesis settings in a .csv file.

In the recordings, the singers were asked to sing on vowels and then glottal fry on the same vowels. The resonances and anti-resonances are easier to locate by taking the spectrum of a single glottal pulse, which makes it easier to match the formant and notch frequencies with these recordings (Fig. 3).

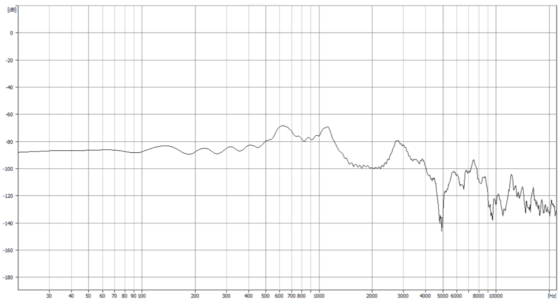
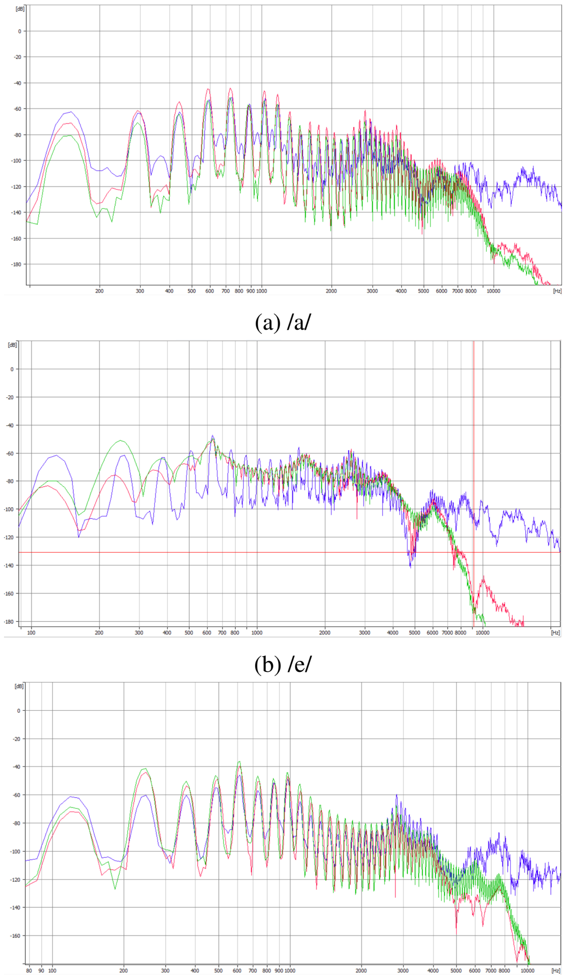


Figure 3: The glottal fry on vowel /a/ from a recording. The formant an notch frequencies are explicitly shown in the spectrum.

For a real vowel singing voice, a synthesis with the same formants and notches and a synthesis with only the formants are created. The spectra of two syntheses are compared with the real one and the results are listened by human to test which synthesis performs better.

For each vowel singing section, the synthesis with formants and notches and another synthesis with only the formants are created. The syntheses and real recordings are analysed in *Sopran*, a sound processing and analysis software from Tolvan Data. Different tracks for different recordings (the real one, the synthesis with notches, and the synthesis without notches) are color-coded. The results of the spectra are shown in Fig. 4.

The envelopes of the real ones and syntheses matches significantly. The red ones also mimic the spectral notches in addition to the formants. From listening tests, the ones with notches sound more natural and more similar to the real voices.



(a) /a/

(b) /e/

(c) /o/

Figure 4: Comparisons in the spectrum. Sound file: amvowels.wav with vowels /a/, /e/, and /o/. The blue curves are the spectra of real voices. The red ones are from the syntheses with both notches and formants. The green ones are from syntheses without notches.

5. CONCLUSION AND DISCUSSION

An improved model for vowel singing synthesis is proposed in this project, with a novel time-reversed source sound model, and a focus on the spectral notches generated from the vocal tract. The synthesis has got a good performance for human ears compared to the real sound samples. However, it is still difficult to keep the higher frequencies in the spectrum. In Fig. 4, an obvious slope from around 8 kHz is observed for every synthesis. This is because of the superposition of many low-pass filters. It remains a challenge to lift the higher end of the spectrum in a source-filter singing synthesis.

6. FUTURE WORK

Currently, the GUI has to be run inside SuperCollider. A future plan is to make a standalone software from the current synthesizer. As SuperCollider supports making standalone applications, the re-implementation shouldn't be very difficult.

Another plan is to apply Linear Prediction (LP) for arranging formant and notch frequencies. Manually tuning the synthesis is time consuming and the usage and application of the synthesis is restricted because of a lack of automation. Machine learning might also be combined to generate a appropriate setting for a synthesis.

This project was going to be a tool for a study in exploring an acoustic aspect of choir singing. A choir singer might find it easier to sing next to one than another in the same section. The hypothesis is that certain spectral features contributes to this phenomenon. And the synthesizer was going to be used as a virtual singer for experiments between groups of a real singer and a mimic. Now that the synthesizer is capable of imitating a human voice in vowel singing, we might apply it in the study as planned.

The method that models both vocal-tract formants and notches might be integrated in the re-implementation of the KTH singing synthesis engine [1].

7. REFERENCES

[1] J. Sundberg, “The KTH synthesis of singing,” *Advances in Cognitive Psychology*, vol. 2, 01 2006.

[2] ——, *Human Singing Voice*. John Wiley & Sons, Ltd, 1997, ch. 139, pp. 1687–1695. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470172544.ch139

[3] S. Ternström and A. Friberg, “Analysis and simulation of small variations in the fundamental frequency of sustained vowels,” *STL-QPSR*, vol. 30, no. 3, pp. 001–014, 1989.