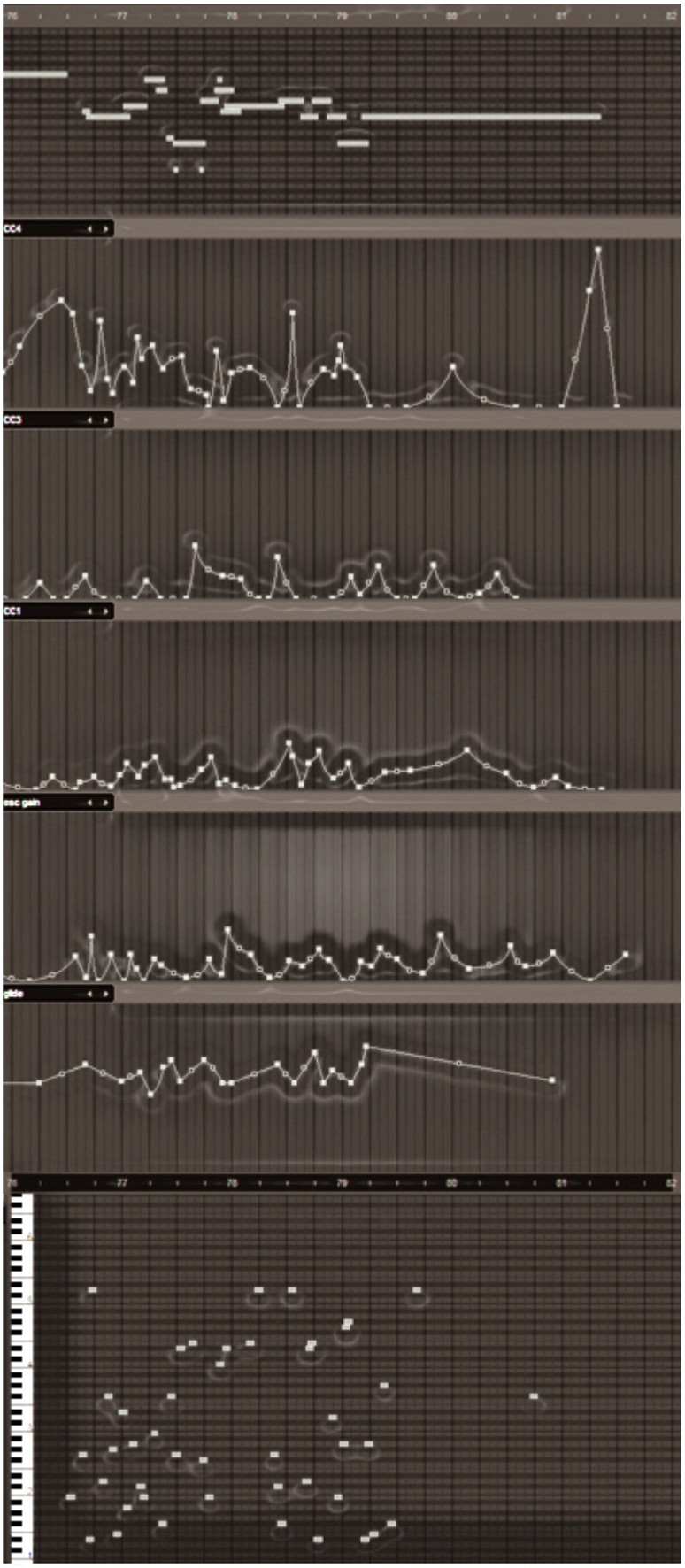
Syng3

voice synthesizer by xoxos



Syng3

introduction

Syng3 extends classic voice synthesis technique to a 7 formant band source filter model, increasing realism and retaining flexibility. The VST continues the functionality of previous versions, sequencing pitch on MIDI channel 1 and phonemes on channel 2.

This VST is presented as a canonical solution for voice synthesis applications. It is implied that the quality of results depends on the ability of the user to elicit them from the form provided - it is not a voice though it can be coaxed to resemble one.

The seven band analysis often extends above 10kHz. To inhibit the use of the model for impersonation of the author over bandwidth limited mediums, the central most region of the formant shift range has been blocked out.

use

Copy the syng3.dll file to your VSTPlugIns folder.

Syng3 requires your VST host to transmit note data on MIDI channel 1. You must also configure a sequence to send note data on MIDI channel 2 to switch phonemes.

signal path

The source component of the model consists of a bandpass filtered noise source and a lowpass filtered oscillator. The oscillator is a bandlimited impulse train, which produces all harmonics at the same amplitude, so that pitch will not affect the amplitude of frequency bands as it would with a sawtooth oscillator.

I have read that the female glottis tends to close less fully, so that on average women have "breathier" voices.

These parameters are located on the top panel of the gui. The other parameters on this panel affect classes of phonemes, and the rate of switching between them. Modulation of these top panel parameters is where the expressivity and personality of your voice will be created.

For effective use, it is not only necessary to understand how phonemes are used in the language you wish to render, but also how I used them in the VST :)

phonemes

The model of speech implemented here divides components of speech, or phonemes, into three classes: vowels, consonants, and diphthongs. Vowels are the conceptually simplest class in the model: sustained tones with a transition time of several milliseconds. Consonants generally have the shortest transition rates of perhaps a few milliseconds. The implementation of consonants is complicated by cases such as plosives. These events are automated by the phoneme switching circuit.

Diphthongs consist of two phonemes and transit between them, such as "ow" or "oi". The "long vowels" are also handled like diphthongs in this model, such as "a" in "save".

Phonemes and sets of phonemes differ between languages, regions and speakers. The set presented here is RP or "Received Pronounciation," perhaps more commonly known as "bbc1 english". While many phonemes are extremely common among human languages, there will be some adaptation required suiting the phoneme set provided to what you may wish to create. For instance, if emulating a western american accent, i would use "ah" as in "far" for the sound "i" as in the speaker referring to themselves, fade most sustained vowels into a schwa, use "d" for "t" in many instances, and so forth.

fascinating historical anecdote

I think most of the programming was done in 2011. The project was shelved because I had built the engine to switch phoneme sets, and in the meantime I started working with FFT, which I thought was a better solution to handle user data. As years went by and I hadn't initiated that project, I thought I ought to make what I already had available.

modulation

The signal path is simple and presents a small number of parameters, it is the compound of the parameterisation that provides character, expressivity, emotiveness, performance.

Even though there are a small number of signal path parameters, elastic animation of several of them can result in a performance that is too volatile to be convincing, given the finite range of credibility we are working with. Separate modulation of several parameters can have what i term a "silly putty" effect. In contrast, if only one parameter were modulated, the perception would be starchy, but retain character more. Between these boundaries is a range that can be exploited to result in better performances, usually or hopefully!

Syng3 is set up to accomodate a variety of modulation schemes. There are several "lettered" knobs on the gui that can be apprehended by the host, then assigned, if your host automation provides smoother modulation than MIDI CC#s.

For simplicity in composition, I recommend assigning a single modulation source as "expressivity" to a group of destinations, like amplitude and brightness. If you become absorbed in realistic or complex emphasis, this approach saves time. I also strongly recommend assigning a modulator for the reduction of amplitude.

don't you know?

The "LCD" displays on the gui are used to assign source and destination for modulation. Editing them is performed by dragging the appropriate region vertically, as if it were a knob.

eliciting the best performance

The range of the model makes it more realistic than other source filter models I have used, but it is still a fairly lofi technology. At the time of writing this I have only begun to explore the particularities the build presents - transitioning from one formant to another isn't guaranteed to sound throatlike. In some cases, perhaps just with one note, I find buffering a transition with an extra phoneme (schwa?) or rate modulation or dip in amplitude helps to concoct a more believable render.

The source was recorded at a relatively strident speaking volume. I am certain the first thing you want to do is add a highpass filter right after it. The model has seven bandpass filters and a lowpass under the lowest band so as not to lose bass, which can sound weird at some pitches with extreme formant shift values.

You may be able to push your timbre to more of a power singing tone by boosting somewhere under 3kHz, the "singer's formant" range which carries during public performance.

Eliciting good results means working with the foibles of the instrument - occasionally a particular transition or combination of pitch and/or phoneme may sound unnatural. Composition means selecting the best and avoiding the worst.

The most defining characteristic of a patch is the shift value. After finding a good value, set the noise filter cutoff and resonance to describe the size and character of the speaker. I perceive character as being the behaviour of parameters beyond these.

After sequencing the pitch, I begin to sequence the phonemes by adding the vowels in quantised mode. Then I opt for fine sequence adjustment and position the consonants slightly before the step positions. Some consonants such as "r" do work better if they overlap into the quantised step. These are very general guidelines and hopefully ignored by those with their inherent sensibilities, though positioning consonants directly on the step is often masked by percussion.

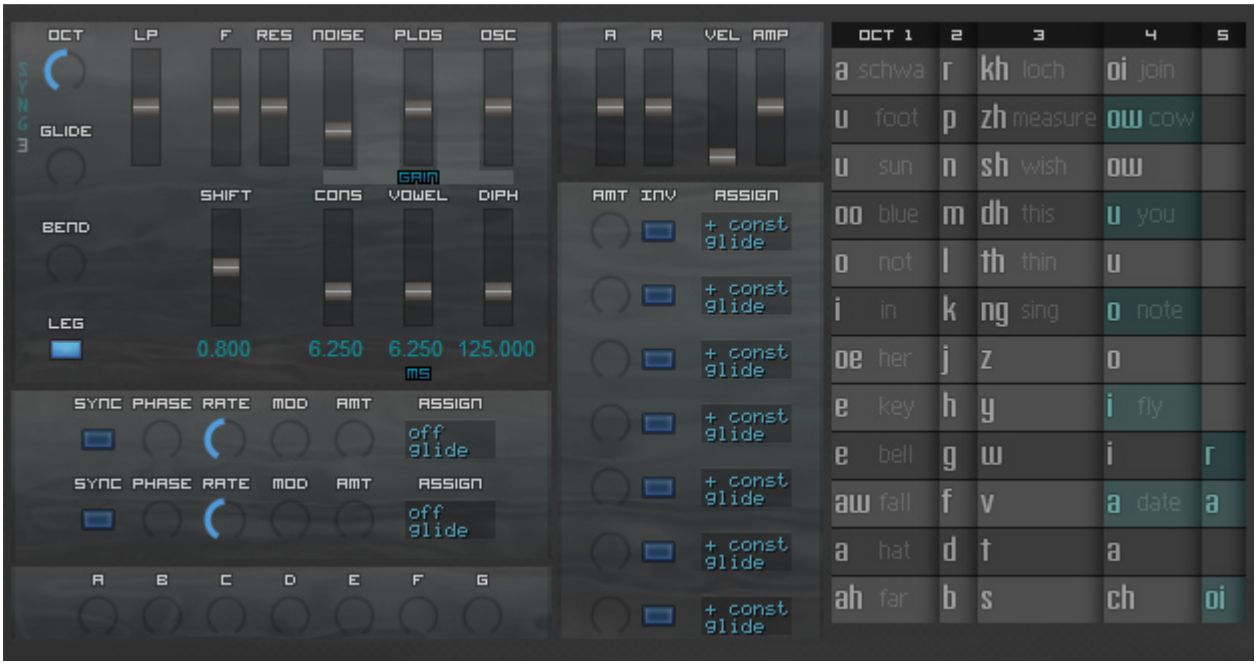
I tend to get carried away adding detail to expressive modulation. I recommend pencilling it in and gradually refining elements in relation to the whole piece. Due to the intense amount of editing that can be put into a synthetic vocal, spending a lot of time focusing on a single measure can develop character that isn't connected to the entire performance.

Proficient use of Syng3 probably requires a foundation of editing vocal recordings and analysing them spectrally, and listening analytically to voice performance until sensibility for calibration is informed. (.....In addition to other assumed proficiencies, eg. practiced phonetics.. caring about what a schwa is..)

The timing of diphthongs (among other things) affects their character, for instance, the long "o" sound, which begins sort of "u" and ends much like "w", is often pronounced with the first sound sustained instead of immediately fading to the "w" sound.

MIDI key D5 on channel 2 triggers the schwa phoneme with a long diphthong fade time. (D#5 triggers "r" with a long fade time.. use them!) Try using this at the end of other vowels to add a relaxed feel to speech. Of course synthetic speech is possible and other effects like laughing with appropriate pitch programming, so synth rap is achievable.

Always begin with amp velocity sensitivity at maximum before you begin drawing notes in the sequencer. After your mix has developed and your expression automations are perfected, note velocity is an easy way to selectively change volume if certain parts need adjustment.



There is no expression of guarantee, please demo before purchase. Syng3 VST was created with the SynthEdit SDK - www.SynthEdit.com

©2015 Rurik Leffanta - All Rights Reserved - VST is a trademark of Steinberg Soft und Hardware GmbH.