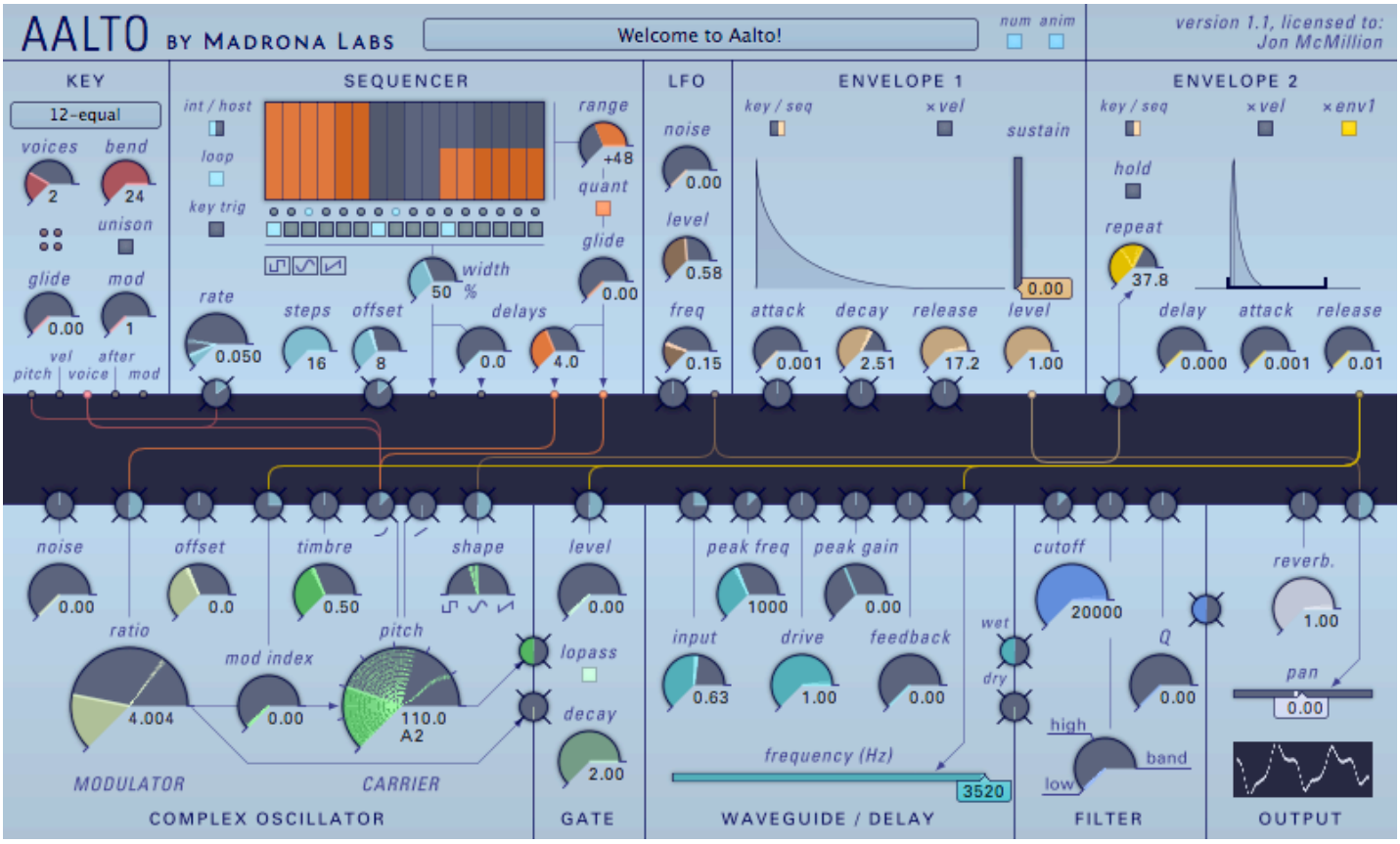


Aalto Quickstart

version 1.1

Welcome to Aalto! This quickstart guide assumes that you are familiar with using softsynths in your DAW or other host program of choice. It explains how Aalto's dial objects and patching interface work, then takes you on a brief tour of each module in turn.

INSTALLATION

To install Aalto, double-click the “Install Aalto Demo” file or, if you have bought the full version, the registered file “Install Aalto for [your name]”. The installer will create directories for presets if needed. Your user presets go in (home)/Library/Audio/Presets/Madrona Labs/Aalto. Factory presets go in /Library/Audio/Presets/Madrona Labs/Aalto.

GENERAL

Aalto's interface has many copies of the same kind of circular control. These controls are both dials that let you control a signal and actual working oscilloscopes that let you see the previous 1/30 second or so of the signal. This scope function is very useful for seeing the effects of modulations.

When multiple voices are active, the dial signal for every voice is shown simultaneously in the dial. This typically appears as several overlapping lines or squiggles.

You can move a dial in the following ways:

Click in the dial's track (the dark area within it) to set the value to the click position. While still holding, drag up and down to fine-adjust the value.

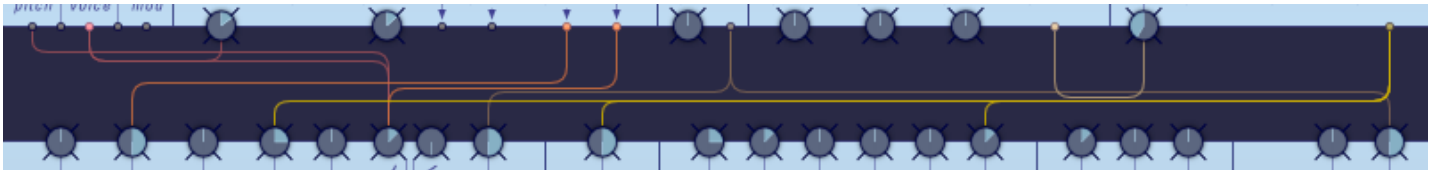
Hover over a dial and use the scroll wheel to fine-adjust the position. Scrolling faster accelerates the adjustment.

Click and drag vertically on a dial outside the track to fine-adjust from the current position.

To return a dial to its default position, double-click or command-click it.

Some dials have logarithmic scales. This was done where a log scale matches the changes you perceive better than a linear one, as in pitch, for example.

THE PATCHER



The patcher is the large dark central area surrounded by all the modules. It lets you patch signals from outputs of modules to inputs of modules.

Signal outputs are the small circles. They light up to show the current value of the signal.

Signal inputs are the very small dials bordering the patcher. These dials do not display signals. They allow you to adjust the amount of signal applied to an input.

To make a patch cable, drag from an output to an input.

You can move either end of a patch cable by dragging the ends around to a different input or output.

You can delete a patch cable by dragging either end to where it is unconnected.

If you create a new instance of Aalto, it will appear with no cables made, so it won't make any sound. Select the "default" preset to add two cables and make a simple sinewave patch.

The input dials control the amount of signal applied to an input. If multiple cables go to a single input, the signals are added together. The sum of all these signals is then multiplied by the input dial value. Most input dials are bipolar, meaning an input dial value can be either positive or negative.

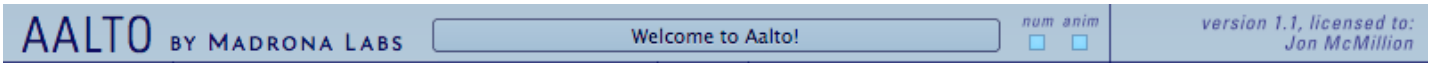
Note that signals don't always flow down: input dials are on both the top and the bottom of the patcher.

Some output signals, such as the envelope outputs, send only positive values. Others, like the signal from the "pitch" output on the KEY module, and the LFO, send signals that swing both positive and negative.

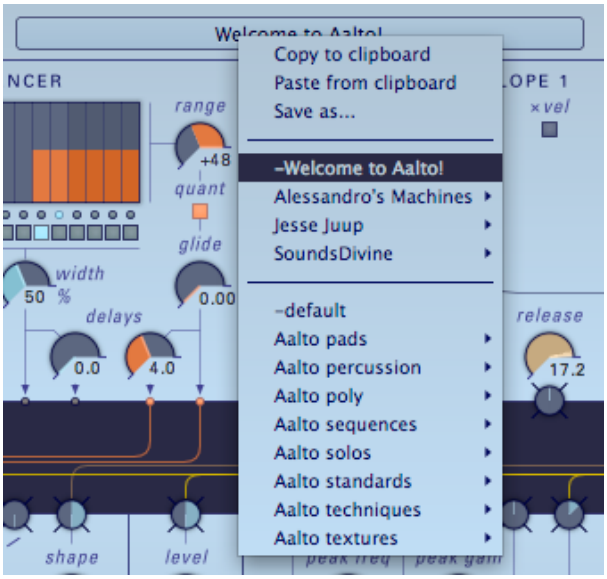
When a patch cord is made, it is made for all of Aalto's four voices.

MODULES

TOP SECTION



The top section contains controls that don't get saved with a patch. The big drop-down menu in the middle shows the current patch name and lets you select patches from a hierarchical menu. The menu is refreshed each time you click on it, so new patches will show up right away after you save them or move them from outside Aalto.



The drop-down menu has three sections. The first section shows the Copy, Paste, and Save commands. When you select "Copy to clipboard," the current patch is saved in a text-only format that you can paste into other text documents. This lets you send a patch to a friend in an email, or post it on a forum, for example. "Paste from clipboard" does the reverse. "Save as..." brings up a file chooser from which you can select a new file to save the patch to, or an existing one to overwrite. When you save a patch form the Audio Units version of Aalto, it goes into the **.aupreset** format. This is a compressed XML format compatible with Logic, Live and other Audio Units applications.

The second section of the menu shows your user presets. These are on your hard disk in <home>/Library/Audio/Presets/Madrona Labs/Aalto. You are encouraged to save your own creations here. Some default user presets are installed by the Aalto installer.

The third section has the factory patches. These are in /Library/Audio/Presets/Madrona Labs/Aalto, and are also installed by the Aalto installer.

Going back up to the top section, the [num] button turns numerical display on and off for all controls. Because maybe you don't want to think about numbers, and maybe you do. The [anim] button turns animations in scope dials on and off. While fun and useful, these animations are somewhat CPU intensive, so you may want them off.
The right top corner shows the version of Aalto you are running, as well as your license info.

KEY

The key module turns all the MIDI data you send Aalto into signals, and makes them available in the patcher.

The menu on the top selects a tuning table. A tuning table maps MIDI notes to frequencies. The default tuning, 12-equal, is the basis for most modern Western music but there are thousands of different tunings to try. These can be loaded from scala .scl files. You can also make your own .scl files using the free software Scala, available at http://www.huygens-fokker.org/scala/.

The [bend] dial selects the pitch bend range in semitones.

The square of lights shows which voices are active at any given time. The "unison" button selects unison mode, in which all the voices are played at once. Legato playing, where one envelope holds over multiple continuous notes, is also active in

unison mode.

[glide] shows the time in seconds that a pitch transition takes between notes.

The bottom of the key module has five different signal outputs: [pitch], [vel], [voice], [after] and [mod].
When the MIDI note C3 is played, the pitch signal output is 0. This has the same result on a patcher input as when nothing is connected. C4, an octave higher, outputs the value 0.25. Each octave adds another 0.25. This scaling was chosen so that keyboard input maps naturally to all the various control signals.

In the patcher, the input dials for OSC pitch, delay peak freq, delay frequency, filter cutoff, and env2 repeat are all set so that when you connect a pitch input and the default scaling (double-click), they will track the same frequencies or intervals.

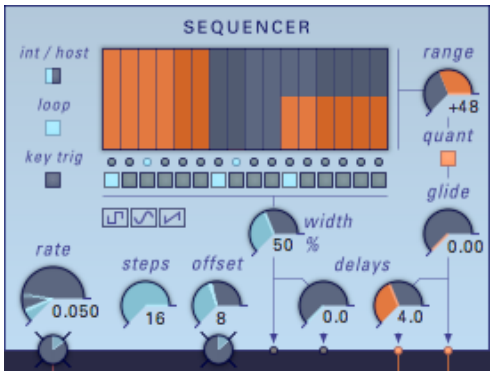
The [vel] output sends a signal proportional to the MIDI note velocity. This signal maintains its value after the key is released.

The [voice] output sends a signal proportional to the number of each voice: 0 for voice 1, 0.25 for voice 2, and so on. This can be used to quickly make changes to the patch that are different for each voice, such as panning all the voices across the stereo field, or setting each sequencer to a different rate. But we get ahead of ourselves.

The [after] output sends polyphonic aftertouch data for each key, added to the channel aftertouch value.

The [mod] dial selects the MIDI controller number that gets routed to the [mod] signal on the patcher. When the number 1 is selected, the modulation wheel is used.

SEQUENCER



The sequencer can make complex shapes over time with its pulse and value outputs. Pulse output controls are blue and value controls are orange. The line of lights is not clickable-- it shows which step is active in each voice of Aalto. The line of buttons control pulse outputs, as on a drum machine.

Each voice of Aalto has its own internal copy of the sequencer that can be set to a different rate and triggered independently. This can lead to some very complex textures very quickly.

The [int / host] button selects a clock source: Aalto's internal clock or the playback clock in your host application.

The [loop] button selects whether the sequencer repeats at the end or not.

When the [key trig] button is pressed, the sequencer

The [rate] dial controls the sequence rate in Hertz.

[steps] controls the number of steps in a cycle. [offset] is added to the current step. After offset is added to step, the sum is wrapped around the 16 steps.

[width] controls the percentage of a pulse during which the pulse output is turned on.

[range] multiplies the sequencer output by a positive or negative value, calibrated in semitones with respect to the 0.25 / octave pitch standard.

[quant] quantizes the output to the current tuning table.

The [glide] knob controls the amount of glide, scaled with respect to the sequencer rate. At a glide of 0, the output steps immediately to the next value on each step. At a glide of 1, the value ramps linearly, reaching the next value at the end of each step.

The [delays] delay the pulse and value outputs by a number of sequencer steps from 0-8.

The three waveform buttons in the center snap the sequence value sliders to the preset shapes shown on the buttons.

LFO

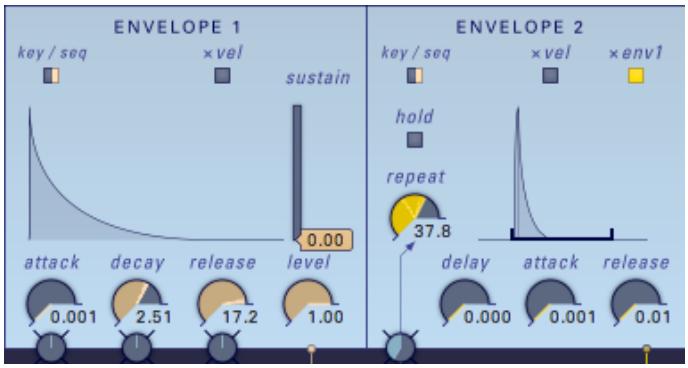
The LFO module is a simple sinewave LFO with a controllable amount of noise faded in.

The [noise] dial fades the output signal from a sine wave to pure noise.

The [level] dial controls the amount of signal sent to the patcher.

The [freq] dial controls the frequency of the sinewave and a filter cutoff for the noise. The filter frequency is equal to the sine frequency multiplied by 50. This value was chosen by ear to create a good balance between sine and noise as modulation sources.

ENVELOPES



Both envelopes have an input selector that chooses between MIDI input [key] and sequencer pulses [seq] as trigger sources. When [seq] is selected, pulses from the sequencer output are sent to envelope 1, and delayed pulses from the second sequencer pulse output are sent to envelope 2.

The [x vel] control on each envelope multiplies that envelope's output by the MIDI key velocity.

The graphs in each envelope module show the actual shape of the envelope over time. They are scaled to match the total duration of the envelope sequence. Envelopes have logarithmic attack and decay curves, and times are calibrated to correspond with the time at which the output value has traveled approximately 60% of the way of its destination.

ENVELOPE 1

Your old friends [attack], [decay], [sustain] and [release] each have controls on envelope 1. In addition, a [level] knob sets the overall output level for additional options when patching.

ENVELOPE 2

as ENVELOPE 1, except:

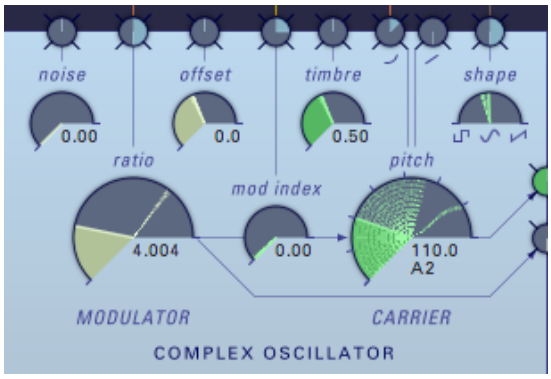
The [hold] button produces the same result as turning [sustain] to either 0 or 1.

An additional [repeat] control re-triggers the envelope the number of times per second that you set. The repeat interval is shown as the dark bracket in the envelope display. [repeat] does not do anything when [hold] is on.

[delay] sets a period of time to wait between envelope trigger and the start of the attack. This is useful for making complex envelope shapes in combination with ENVELOPE 1.

Selecting the [x env1] button multiplies the output of ENV2 by the current value of ENV1.

COMPLEX OSCILLATOR



The complex oscillator is the heart of Aalto. It contains two internal oscillators: **modulator** and **carrier**. This is two-operator FM, with an interface and sounds inspired by the wonderful Buchla 259, with some twists.

The modulator oscillator has three controls:

[noise] controls its noisiness. This actually fades between a sine wave and noise that is bandpass filtered to match the sine wave frequency.

[ratio] controls its frequency as a ratio to the carrier oscillator.

[offset] adds a constant offset in Hz to the frequency. To get a constant frequency, set [ratio] to 0.000 then dial in a Hz offset.

The [mod index] controls the amount that the mod oscillator modulates the frequency of the carrier oscillator. Control signals are sent to the mod index through a multi-stage lowpass filter that mimics the response of an optoelectronic isolator (Vactrol).

The [timbre] dial controls a complex wave folder network that mimics the Buchla 259 wave folder. In order to avoid aliasing, the effect of the timbre dial gets less strong as the oscillator frequency is raised.

The [pitch] dial controls the rest pitch, or the pitch produced when you plug in a signal from the KEY pitch output and hit MIDI key A3.

Unlike other dials the pitch dial has two patcher inputs: an exponential one and a linear one. The exponential one is calibrated to match the 0.25 / octave standard of Aalto's signals. The linear one has a much wider range and can be modulated for fast "thwips," inharmonic sounds and other effects.

The [shape] dial uses an antialiased waveshaper to fade between a sine in the center, a square on the left, and a saw on the right.

The oscillator module has two small dials at right that select the amount of the carrier oscillator and the modulator oscillator to send to the GATE module.

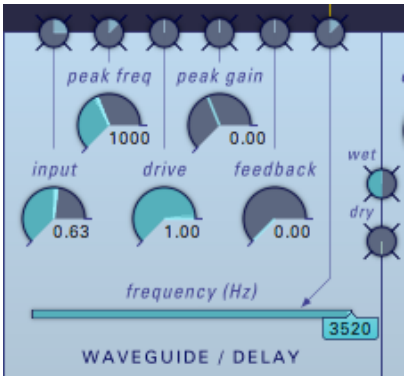
GATE

The [level] dial is the main way to make an enveloped sound, typically by patching in a signal from ENV1 or ENV2. But since Aalto's oscillators are always running, you don't need to play a MIDI note to make sound – turn the level control up to start your drone exploration!

The [lopass] button puts the gate module in lowpass filter mode. When [lopass] is off, GATE is a simple volume control.

[decay] controls the decay of another Vactrol-like filter that filters the control signal. When [decay] is 0, the control is unfiltered. When [decay] is at 2, the decay is at maximum.

WAVEGUIDE / DELAY



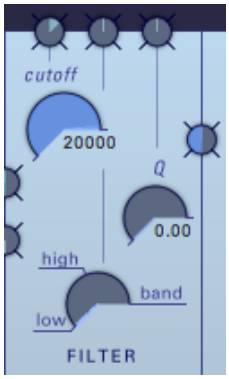
This module is an allpass-interpolated delay with a waveshaper and a peaking EQ built into the feedback loop. Because it has such short and controllable delay times, unlike a typical analog delay, it can be used as an additional oscillator or waveguide.

[input]: input to delay
[peak freq]: frequency of peak filter inside delay
[peak gain]: gain of peak filter inside delay
[drive]: distortion amount inside delay
[feedback]: feedback feedback feedback
[direct]: sends direct signal through to output, added to delay signal
[frequency (Hz)]: repeat frequency. Logarithmic scale. Patching the pitch input to this slider controls it in musical intervals that match the osc pitch.

A tiny bit of noise is added to the input of the delay, so that either lots of drive or feedback or both will put the delay into self-oscillation. Of course the most interesting sounds happen right at the verge of oscillation.

The wet and dry output dials control the amount of delayed signal and direct signal from the gate mixed together at the output.

FILTER



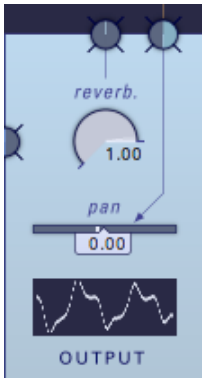
Modeled after the Oberheim SEM filter. A state-variable filter with mixable simultaneous outputs.

[cutoff]: a logarithmic dial. Again, patching the pitch signal in here gives consistent pitch intervals.

[Q]: from 0 to just below self-oscillation.

mix knob: controls the mix of signals from lowpass, to highpass, to bandpass.

OUTPUT



[reverb]: a reverb with no controls, inspired by the reverb tank in my Arp 2600. The knob controls the amount of input from each voice to the reverb.

[pan]: modulates the pan of a signal between left and right outputs. Can be modulated at audio rates for crazy spatial effects!

Oscilloscope: shows the level of the output signal.

Have fun and if you have questions, please ask on our forums!