Sound with a soul
Voxengo

Voxengo Boogex User Guide

Version 3.6

https://www.voxengo.com/product/boogex/

Contents

Introduction

Boogex is a guitar amplifier plug-in with a variety of sound shaping features, for professional music production applications. With Boogex it is possible to get a heavy distorted sound as well as slight "jazzy" saturation sound. Boogex is also able to apply any speaker cabinet impulse response (selection of built-in impulses is available). The processing latency is close to zero making it possible to use Boogex for real-time guitar processing.

Boogex also includes the input gate module, and reverberation module derived from Voxengo OldSkoolVerb reverb plug-in.

Boogex produces a nice "minimalist" rock music-geared sound which may be a bit noisy at higher overdrive settings; higher frequencies can be easily suppressed with its built-in Emphasis EQ.

Boogex can be also used as a plain stereo convolution processor when its "Amp" stage is turned off. The convolution module has zero latency and is not CPU-demanding. With the amp modes Boogex currently provides, it is not particularly well-suited for metal music, but can still be used as a cabinet impulse response processor for metal music.

Features

- Emphasis EQ
- 2 amplifier types
- 14 amplifier modes
- 61 built-in cabinet impulse responses
- DynaCab processing
- Stereo convolution processing
- Gate module
- Built-in stereo reverb
- 64-bit floating point processing
- Preset manager
- Undo/redo history
- A/B comparisons
- Contextual hint messages
- All sample rates support
- Zero processing latency

Compatibility

This audio plug-in can be loaded into any audio host application that conforms to the AAX, AudioUnit, VST or VST3 plug-in specification.

This plug-in is compatible with Windows (32- and 64-bit Windows XP, Vista, 7, 8, 10 and later versions, if not announced otherwise) and macOS (10.11 and later versions, if not announced otherwise, 64-bit Intel and Apple Silicon processor-based) computers (2.5 GHz dual-core or faster processor with at least 4 GB of system RAM required). A separate binary distribution file is available for each target computer platform and audio plug-in specification.

User Interface Elements

Note: All Voxengo plug-ins feature a highly consistent user interface. Most interface elements (buttons, labels) located at the top of the user interface are the same in all Voxengo plug-ins. For an in-depth description of these and other standard features, and user interface elements, please refer to the “Voxengo Primary User Guide”.

Emphasis EQ

Emphasis equalizer is used to shape the sound of the amplifier. This is a very powerful tone-shaping stage. Please refer to the “Voxengo Primary User Guide” for in-depth information about this EQ control surface’s functions.

Emphasis EQ allows you to overdrive certain areas of the spectrum more or less than the others. Note that this is not a usual equalizer: the bands produce overdrive tone adjustments, and not the equalization effect.

The low- and high-pass filters are applied both before and after the amplifier module.

Pre EQ

Used to filter the incoming signal. This is usually useful if the pickups on the guitar are too bassy or too crisp. This EQ can be also used to shape the overall sound.

You can enable the “Gate” module to enable input signal gating (which is applied before the Pre EQ stage). The slider adjusts gate’s threshold level.

Amp

Here you can select the amplifier type and mode. Note that the difference between different modes of the same type may not be pronounced. The “Type 2” amp mimics a 3-stage valve amplifier.

The “Drive” knob specifies the amount of amplification in decibel.

By means of the “Pre EQ Mix” knob (in percent) you can mix in the input (gated and pre-equalized) signal to the distorted and emphasized signal before it goes to the speaker cabinet convolution processing.

Note that when the “Amp” stage is disabled, the convolver works in stereo mode. In this case, the “Pre EQ Mix” knob should be set to “100” to additionally bypass the Emphasis EQ stage, or left at lower values if the low- or high-pass filtering is desired.

Cabinet Sim/Convolver (Mono, Stereo)

Speaker cabinet impulse response selects which speaker cabinet model (and mic) to use. It is possible to load an external audio file (WAVE, Wave64, AIFF), but you have to be careful with large files as they may overload your CPU quickly.

The “X” button unloads a currently loaded audio file, and switches to the internal impulse response selector.

The “Amp Mix” knob (in percent) specifies the amount of dry amplified sound to blend with the convolved sound.

The “Dyn” switch enables the DynaCab processing.

DynaCab Editor

When the “Dyn” switch is enabled, the “DynaCab” button becomes available. This button opens the “DynaCab Editor” window which controls the DynaCab processing. Note that this window does not contain presets, because DynaCab requires an elaborate fine-tuning suitable for the input signal.

The DynaCab processing is simple in its nature: the Amp signal is processed by two independent convolution processors, each loaded with its own impulse response. Then the outputs of these convolution processors are mixed in a dynamic manner. The dynamic response is derived from the signal output of the Pre EQ stage. This preserves the original guitar playing dynamics. You may take a look at the displayed graph which oscillates between the “Cab 1” and “Cab 2” points: this graph shows which cabinet is currently prevalent in the output signal.

The DynaCab tuning is best started by choosing an appropriate threshold: the “Cab 1 Thrs” selects the Pre EQ stage signal’s upper (peak) threshold level which corresponds to full “Cabinet 1” output. The “Dyn Range” then selects the negative delta of Pre EQ stage signal’s level which corresponds to full “Cabinet 2” output. So, when the signal is loud, the Cabinet 1’s response will be prevalent in the DynaCab’s output, but when the signal is quiet (lower by the “Dyn Range” decibel), the Cabinet 2’s response will be prevalent instead. At intermediate Pre EQ stage’s levels, both cabinets will be mixed in a varying proportion. The “Dyn Range” should be chosen according to the guitar playing dynamics.

The “Attack” and “Release” parameters affect envelope detector’s timing characteristics. For snappy response, it is suggested to use lower values for both parameters.

Note that impulse responses of both cabinets are loudness-matched by default. The “Cab 1 Gain” parameter allows you to boost or cut the loudness of Cabinet 1’s output signal. In practice, this works as an expander effect on gain boost, and as a compressor effect on gain cut. The “Cab 1 Delay” parameter time-shifts the Cabinet 1’s output by the specified amount in milliseconds. Such delay allows you to better align both cabinet impulse responses to reduce comb-filtering artifacts.

You can load any impulse responses in the “Cabinet 1” and “Cabinet 2” selectors – even reverb responses, for special sound effects. In an ideal case, both impulse responses should be matched: e.g. by capturing two impulse responses from a real guitar cabinet, at high and low loudness levels. Non-matched impulse responses can be used for special sound effects, but finding a good match may be a tedious experience.

The “Topology” switch changes the DynaCab processing’s topology. While the “Post” topology is straight-forward: output signals of two convolution processors are mixed dynamically, the “Pre” topology may be harder to visualize. In the “Pre” topology, the “Amp” signal is, at first, dynamically split into two paths, with each path then fed to its own convolution processor. Then the output signal of two convolution processors is simply summed. The difference between the “Pre” and “Post” topologies is most apparent on reverb impulses: for example, if the “Cabinet 1” is loaded with a short reverb while the “Cabinet 2” is loaded with a long reverb, louder input signals will excite a shorter reverb while quieter signals will excite a longer reverb. With the “Post” topology such reverb arrangement sounds different: the output signal is switched between two reverbs without excitation effect. The “Pre” topology usually sounds “fluid” while the “Post” topology sounds “punchy”.

Stereo Reverb (OldSkoolVerb)

This group of knobs affects reverb’s subjective spatial image.

The “Pre-delay” parameter specifies reverb’s pre-delay time (in milliseconds). Imitates distance from the listener to the performer. Lower values produce denser early reflections.

The “Space” parameter specifies imaginary time (in milliseconds) between reflections: in effect, it specifies room’s dimensions. Extremely low values produce “plate reverb” sound and a denser reverb tail. Higher values produce hall reverb sound and a sparser reverb tail. Higher values also produce a more spacious, “transparent” reverb sound, suitable for application over the full mix.

The “Time” parameter specifies reverb’s RT60 time (in milliseconds), the time it takes for the reverb loudness to fall down by 60 decibel. This parameter models both room’s size and overall damping.

The “Width” parameter specifies reverb’s stereo-width (in percent). This parameter imitates room’s width at listener’s position.

The “Gain” knob adjusts reverb’s loudness (in decibel).

Credits

DSP algorithms, internal signal routing code, user interface layout by Aleksey Vaneev.

Graphics user interface code by Vladimir Stolypko. Graphics elements by Vladimir Stolypko and Scott Kane.

This plug-in is implemented in multi-platform C++ code form and uses “zlib” compression library (written by Jean-loup Gailly and Mark Adler), “LZ4” compression library by Yann Collet, FFT algorithm by Takuya Ooura, filter design equations by Magnus Jonsson and Robert Bristow-Johnson, VST plug-in technology by Steinberg, AudioUnit plug-in SDK by Apple, Inc., AAX plug-in SDK by Avid Technology, Inc., Intel IPP and run-time library by Intel Corporation (used under the corresponding licenses granted by these parties).

Impulse responses by Murray McDowall and Nic Beamso.

Happy Mixing!