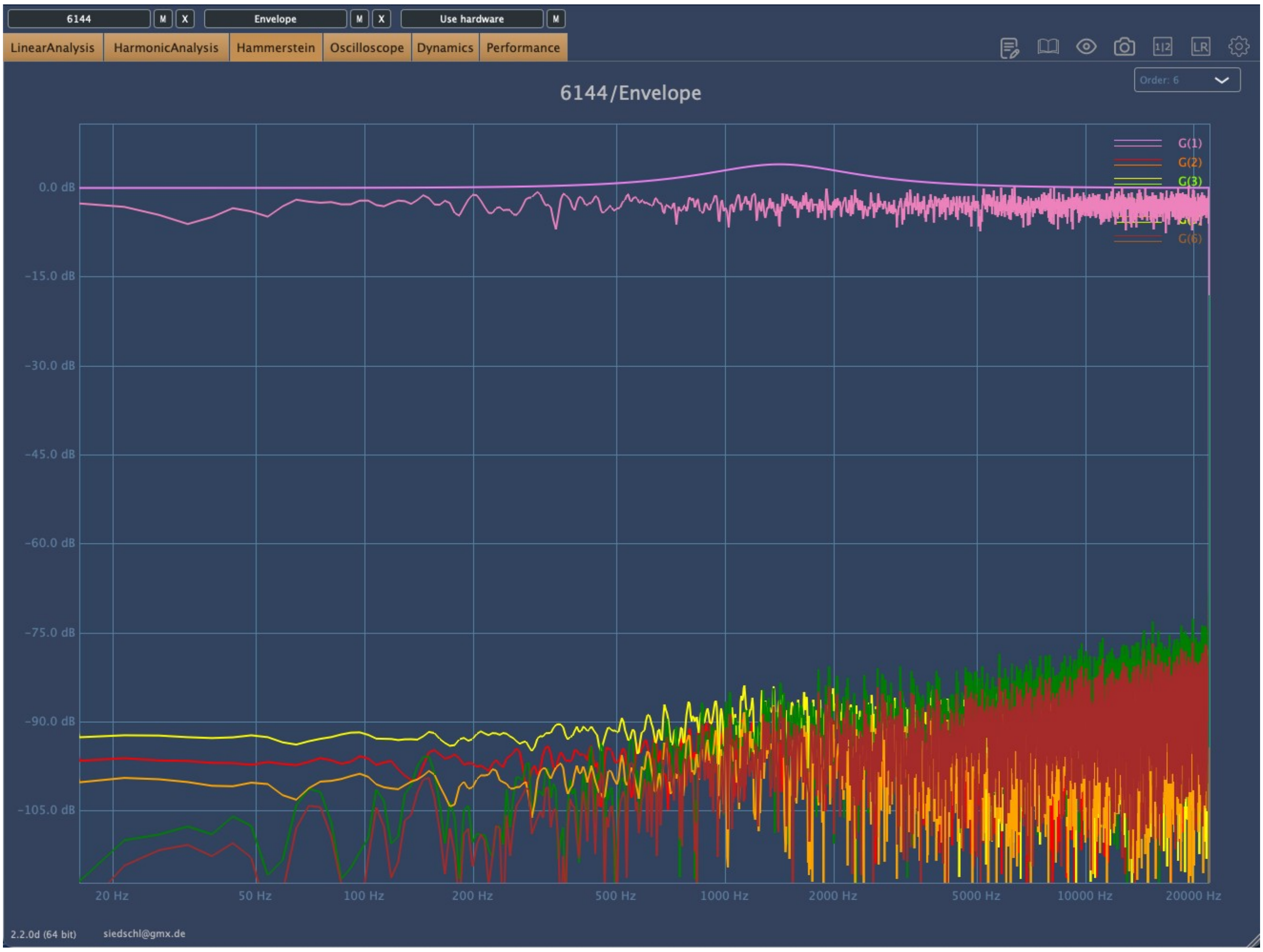
Plugindoctor manual (v2)

Thanks for using Plugindoctor! While the user interface is hopefully designed in a way that the basic usage is more or less self-explaining, this manual should provide you with some more detailed info on what is actually being analyzed in all the different subsections. It is assumed that the reader has a basic understanding of digital signal processing; expressions like "phase response" or "harmonic content" will be used without further definition.



General usage:

Since v2, Plugindoctor can be installed as a standalone app, but also as an audio plugin (in VST, VST3, AU and AAX format). Both versions can load VST, VST3 and AU plugins (OSX only for the latter, of course). AAX plugins can NOT be loaded, due to legal restrictions. In the standalone version, Plugindoctor can also be used to add external hardware devices, see below for more detail. You should find separate 32 and 64 bit versions of the standalone app after installation, which can be used to analyze 32 and 64 bit plugins, respectively. The plugin versions will be available in 32 and 64 bit format by default. Please note that from OSX 10.15 or higher, only the 64 bit version will be installed on your Mac.

Two plugins can be loaded simultaneously, for easy comparison between different algorithms. You can select whether you want to see the results from each plugin independently, or the difference between the two plugins. Whenever you load a new plugin, the first empty plugin slot will be filled first. If both slots are already full (i.e. if you are already analyzing two plugins), you will be asked which of the two should be unloaded and replaced with the new plugin.

When using the plugin version of Plugindoctor (PD), you can analyze plugins in the same way as you are doing it in the standalone version, but at the same time you can listen to whatever a specific plugin is doing to your actual material that you are sending through it in your DAW. If no plugin is loaded at all in PD's plugin version, the DAW audio will just go right through it. If you have two plugins loaded in the plugin version of PD, you will see the results of the measurements of both plugins simultaneously, but only one of the plugins will be used at a time to receive the audio that your DAW is sending. In the plugin version, you can select which of the two loaded plugins is being used for your DAW material by toggling "Play #1" or "Play #2" in the top row of PD's UI. That option will only appear if you actually have two plugins loaded. That way you can quickly A/B two plugins (or two instances of the same plugin, but with different settings) and hear the difference while inspecting what the plugin is doing in Plugindoctor.

Plugins can be loaded either directly from disk, or via a plugin browser. The latter is shown by clicking on the library/book symbol in the upper right corner. Initially, it will be empty. You can use either the "Scan" button to perform a global scan of all supported formats (recommended to quickly populate the plugin browser) or make more specific scans via the "Options" menu, which also allows you to remove selected plugins from the list. Once you have a list of plugins in your browser, simply load one by selecting it and clicking the "Load plugin" button, or by double-clicking on the plugin name in the list.

Alternatively, if you want to load a plugin directly from a location on your harddisk, use one of the two "Load plugin (file)" buttons. Either way, once you have selected a plugin for analysis, its name will be displayed in the selected slot in the top row (#1 or #2), and its window will pop up, and the analysis signal will be routed through the plugin's in- and outputs. Analysis can be interrupted for each of the slots by clicking on the „M" button next to the plugin's name in the top row. To remove a plugin from analysis, click the „X" button.
The plugin window can be closed by either clicking the close button on the plugin window or the eye symbol in the upper right corner of Plugindoctor. A second click on the eye symbol will display a closed plugin window again. If two plugins are loaded, clicking the eye symbol will close all open plugin windows if there are any, or reopen both of them if both are closed.

The camera symbol lets you take a snapshot of Plugindoctor. A file browser will pop up, which will let you choose the location and the name of the generated PNG file.

The button to the right of the Camera button lets you switch between parallel and substracted serial operation when you have two plugins loaded. When the button shows 1|2 , the output of both plugins will be shown in separate curve. When clicking on the button, the display on the button will change to 1-2, which means that you will now only see a single curve, which is obtained by first sending the analysis signal through plugin #1, then, phase-inverted, through plugin #2. This mode lets you easily dial in a close match between two plugins.

The LR/MS button lets you switch the analysis mode between Left/Right and Mid/Side. Many plugins offer a dedicated Mid-Side mode, or are constructed as Mid-Side plugins in the first place. In Mid-Side mode the two curves you will see in most of the analysis windows represent the mid and the side signal, whereas in Left-Right mode the two curves will represent the left and the right channel (provided your plugin is not a mono plugin, in which case there will be nothing displayed in Mid-Side mode because that approach requires a two channel plugin).

Finally, the settings symbol will open up the "Settings" window (also explained in more detail below).

Zooming and changing axis scale

You can zoom into all graphs except in "Performance". Zooming can be done in two ways: first, you can select a rectangle with your mouse, starting in the upper left corner, will draw the selected rectangle while dragging the mouse in the lower right direction, and zoom into the corresponding regions upon releasing the mouse button. Zoom out again by clicking and dragging into the upper-left direction.
Second, you can also zoom and move the whole graph using your mouse wheel. Place the mouse cursor to the left of the graph to affect the y-axis, or to the bottom of the graph to affect the x-axis. Simple mouse wheel scrolling will move the graph along the desired axis, while holding down the CTRL button will zoom in or out along the axis. To reset everything, drag a rectangle from lower left to upper right corner using your mouse again.

The different modes of analysis

Plugindoctor has a range of different analysis modes which can be accessed via the orange tab buttons. Those modes are explained in detail below.

Linear analysis

This section features two modes of input signals for the analysis ("Delta" and "Random"), and two modes of analyzing a plugin's response ("Magnitude" and "Phase"). "Delta" refers to the input signal being a so-called delta peak, which

means the first sample is (usually) equal to 1, all the other samples are zero. This input signal contains a flat contribution of all admissible frequencies, i.e. if you would send this signal through a frequency analyzer, you'd get a flat curve when looking at the magnitudes of the contributing frequencies, and also a flat curve when looking at the phase response. This means that all deviations from a flat curve come from the plugin being analyzed.
The "dB" slider lets you adjust the height of the delta peak, where a value of 0 dB corresponds to a digital sample value of 1.
"Random" will send white noise with a maximum peak level of 0 dB through the plugin. This signal has a flat frequency response too, but only if you average it over a sufficiently long time. The phase response, on the other hand, is flucuating randomly.
Concerning the analysis of the output signal, "Magnitude" will display the absolute values of the contributing frequency components in the Fourier-transformed output of the plugin being tested, while "Phase" will display the corresponding phase response. The default FFT size is 16384 samples. This size can be increased twice- or fourfold in the "Settings" window (see discussion further below).

In the linear analysis window, a curve can be stored pressing the "Store" button and will then continue to be displayed when the loaded plugin's parameters are being changed or a new plugin is being loaded, for easy comparison between settings or plugins. "Clear" will remove this stored curve again.

Harmonic analysis

The aim of looking at the harmonic response of a plugin is to see the response of a plugin when it is being fed with a signal of precisely defined frequency content (as opposed to the delta peak from the linear section).

a: Single frequency mode

Here we have the choice between two different methods of harmonic analysis: "THD" (total harmonic distortion) and "IMD" (intermodular distortion). In THD mode, the input signal consists of a pure sine wave (the frequency and intensity of which can be changed in the "Settings"). The amplitude and frequency of the input wave can be adjusted using the horizontal "dB" and "Hz" slider, respectively. The output will typically consist of a large peak at the input frequency plus, if the plugin generates harmonic content due to a nonlinear algorithm, one or more peaks, typically at integer multiples of the input frequency (higher harmonics). When aliasing occurs, higher harmonics that are theoretically higher than the Nyquist frequency are being produced, which will be reflected off the Nyquist frequency's location and lead to additional peaks at non-integer values of the input frequency.
Apart from visually inspecting the resulting signal, two measures of the distortion generated by the plugin are being displayed in this mode, which will be shown in the upper right corner of the graph: THD and THD+N. THD is

defined as the inverse ratio of the magnitude of the input sine signal and the summed magnitudes of the peaks at integer multiples of the input frequency. THD+N is the inverse ratio of the magnitude of the input signal and the magnitude of "everything else", which includes all higher harmonics but also everything in between.

"IMD" mode uses two input frequencies, a low and a high one, with an intensity difference of 12 dB, i.e. the lower frequency comes in a 0 dB, the higher at -12 dB. The lower input frequency is set to 60 Hz, the higher frequency to 7000 Hz. If the plugin is creating additional harmonic content, this will now not only show up in higher harmonics of the two input frequencies, but also in modulated peaks of the higher frequency by the lower one: you will see a peak at 60 Hz, a peak at 7000 Hz and, if there is intermodular distortion, several peaks at 7000 +- N*60 Hz. Plugindoctor calculates a numerical value for IMD by dividing the RMS-summed contributions of 10 modulated peak lower than 7000 Hz and 10 modulated peak higher than 7000 Hz (all peaks with a 60 Hz distance between 6400 Hz and 7600 Hz) by the magnitude of the incoming 7000 Hz peak.

b: Sweep mode

- *1D:* In this mode, the reponse of the tested plugin is measured over a whole range of excitation frequencies: toggle the "sweep" button and Plugindoctor will continuously sweep from low to high frequencies, with an amplitude you can choose with the "dB" slider. You will note that the button that, before toggling the "sweep" button, was used to switch between THD and IMD, now lets you choose between THD and fundamental. Set it to THD, and Plugindoctor will calculate the THD response per frequency. Set it to fundamental, and you will see the gain or reduction in dB of the currently applied single frequency wave. For strictly linear plugins, this reponse will be identical to the results from the "Linear" section; as soon as nonlinearities come into play, the two measurements are usually different. The lowest frequency is defined by having to fit 10 oscillations into the audio buffer that's used for measurement, so it will be influenced by both your sample rate and your quality settings (the latter is changing the buffer size). Please note that, for the THD calculation, the results will go to zero as the sweep frequency approaches ¼ of the sample rate, as there will be no more higher harmonics left between the applied frequency and the Nyquist frequency (½ the sample rate).

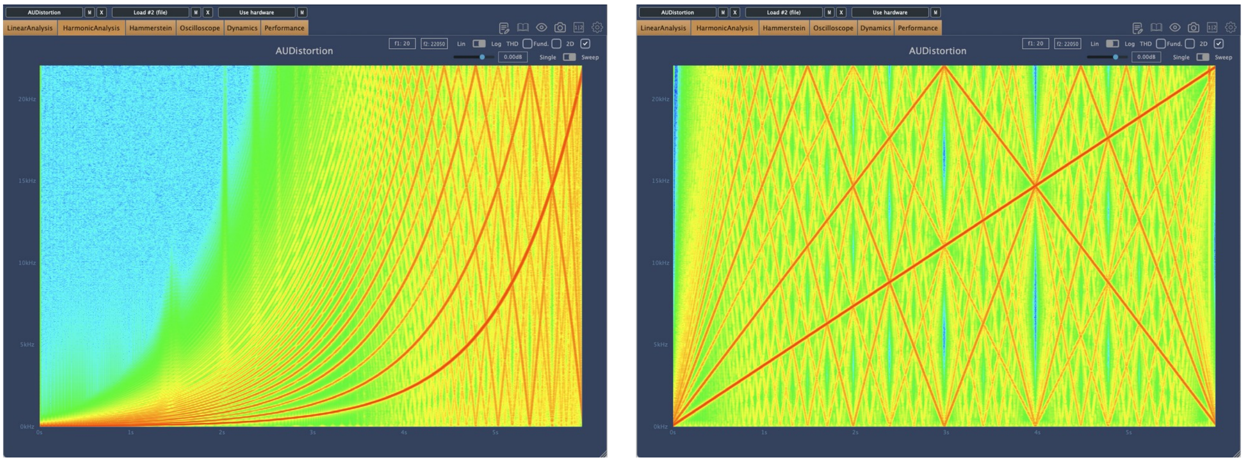


Figure 1: 2D sweep with exponential (left) and linear (right) sine frequency increase, and clearly visible harmonics at integer multiples of the fundamental frequency.

- <u>*2D:*</u> In the two-dimensional sweep scan, the test signal is also a (roughly 6 seconds long) sine sweep, but the result is plotted both as a function of time (along the x axis) and of frequency (along the y axis). To be precise, the two-dimensional colour plot shows the log-abs of the Fourier transform of a series of time-shifted and time-windowed responses of the tested plugin as the applied test frequency is increased. You should usually see the fundamental frequency as a red line, and depending on the plugin producing higher harmonics, you should see one or more lines above the fundamental frequency. The most interesting part is that you can use the 2D plot to quickly spot aliasing: any of the additional frequency lines that hit the upper border of the frequency spectrum (defined by the Nyquist frequency) and then gets reflected when the fundamental frequency is increased further is a sign of aliasing. Please keep in mind that the colour scale is logarithmic here: there is a difference of (by default) -200 dB between the highest (red) colour and the lowest (blue) colour. You can set the maximum and minimum values which are mapped to red and blue with the two-valued slider to the right of the graph area. You can select the low (f1) and high (f2) frequency of the sweep, and switch between linear and exponential frequency increase using the Lin/Exp switch. Only one plugin at a time is displayed here, if you have two plugins loaded (or are using external hardware), you can select which of the results you want to see from a drop-down menu.

Oscilloscope: the oscilloscope window displays the audio signal coming out of the plugin, with a sine wave as input signal. This mode is useful for precisely looking at possible deformations of a sine wave due to, e.g. a very fast compressor or a distortion plugin. There are two ways to display the audio data: when "Time" is selected using the button in the upper right corner, the x-axis will display the time in seconds, while the y-axis shows the audio signal coming out of the plugin. When "Wavesh." is selected, the x-axis corresponds to the audio signal going into the plugin, while the y-axis corresponds to the outgoing signal again. This mode is useful to see how the plugin shapes the incoming audio. In waveshaping mode, a slider can be used to adjust the delay in samples between the in- and outgoing audio data on display. It is set to the latency that's reported by the plugin per default, but delays can occur even with a plugin that officially has a latency of zero samples, due to nonlinear phase effects. Amplitude and frequency of the test signal can be adjusted using the two respective sliders.

Dynamics looks at a plugin's reaction to varying input levels. Again, the input signal is a single sine wave whose frequency can be changed in the settings. There are two modes here: "Ramp" and "Attack/Release". In "Ramp" mode, the plugin is sequentially fed a sinewave (of selectable frequency) with an input level increasing (by default) from -100 dB to 0 dB in steps of 1 dB, and then returning to -100 dB in an endless loop. Using the respective editable labels in the Dynamics tab, you can change the default ramp dB parameters. You can also adjust the time that Plugindoctor takes for each measurement, selecting from 4 values between 0.4 and 1.5 s. While for plugin testing the actual time that you have to wait for a measurement to complete will be much shorter because a plugin can operate much faster than real time, the story is different when testing external hardware, as in that setting, time cannot be sped up. Please make sure that the selected time is longer than any potential attack times, since otherwise the readout will be inaccurate. For each input level, the maximum amplitude of the plugin's output signal is recorded and displayed as a graph.
In Attack-Release mode, the plugin is being fed a sequence of three sine wave with three different levels. The lengths for which each of the three waves is present, as well as their levels, can again be changed in "Settings". Typically, one would look at a certain time span with an input level below a compressor's threshold, then a certain time span above the threshold, and again a third time span below the threshold. By looking at the displayed output signal of the plugin that is being fed this type of signal, one can see how fast a compressor reacts when the signal exceeds its threshold ("attack") or when the signal falls back below the threshold ("release").

Hammerstein: this tab provides a more detailed look at the nonlinear behaviour of a plugin or a piece of hardware. Briefly speaking, the Hammerstein model tries to fit the system under investigation with a power series, with each component of the series multiplied by its own linear filter. You can select the order of the power series, with a maximum of 7 allowed.
That way you can see not only how much a certain higher harmonic (say, x^3) is contributing to the signal, but also how its response is different across different frequencies.
Technically, the Hammerstein model consists of the diagonal terms of the corresponding Volterra kernel. This manual is not the right place to provide a more detailed explanation, I'd kindly refer you to https://ant-novak.com/pages/sss/ for a more detailed explanation of the underlying analysis method.

Performance: this module measures the time spent in a plugin's processing callback as the number of samples the plugin is processing. There will be some fluctuation due to the varying CPU load of your computer, but generally it will give you a good idea whether your plugin compares favourably with other plugins doing comparable processing tasks, or whether there are specific sample sizes which take more time than they should (maybe due to reallocations).

The **Settings** page (which you can reach via the button with the standard settings symbol in the upper right corner of Plugindoctor) lets you control a number of values that are being used in the analysis and when displaying the results:

- the first row lets you select the overall sample rate
- in the next row, "quality" refers to both the global buffer size, and the FFT size being used in "Linear" and "Harmonic". "Normal" corresponds to 16384 samples, with "Higher" and "Highest" increasing this number by a factor of 2 each.
- the "Speed" dropdown menu lets you adjust the processing speed: by default, Plugindoctor shoots as many samples per time to the plugin as possible, which means it is way faster than realtime. This might sometimes lead to an unusual flickering on the UI of some plugins, and of course also to a high CPU consumption. For this reason, you can slow Plugindoctor down: a bit counterintuitive, "Realtime" is the slowest option where the plugin gets exactly as many samples per second as dictated by the sampling frequency. From there, you can increase the speed in three steps, until "ultra" is again as fast as possible.
- You can select between three slightly different background colour schemes using the following three buttons.
- The next six buttons (3x2) let you choose the colours for the various curves PluginDoctor will display (the text on the buttons is hopefully self-explaining)...
- "Load last plugin on start" will let you start a new PluginDoctor session with the same plugin(s) loaded as when you last closed PluginDoctor.
- "Save as default" makes Plugindoctor use the currently selected settings as default.
- If you want to analyze external audio hardware (available in the standalone version only!), please click the "Show audio hardware settings" button to properly route your signal to and from the external device.

Hardware analysis (standalone version)

To analyzer external audio hardware like an outboard EQ or compressor, you need to attach the hardware gear to your audio interface, with one output going from the interface to the hardware's input (can be mono or stereo) and the output from the hardware going back to an input channel of your audio interface. In Plugindoctor's settings, use the "Show audio hardware settings" dialog to select the in- and output of your interface that you have used to connect the piece of hardware. On Windows, you can also choose which driver

you want to use: DirectSound, Windows Audio or ASIO (if your device supports the latter). Please make sure to use the same sample rate in the hardware settings dialog that you've also chosen in Plugindoctor's settings, otherwise you'll get incorrect results.
Once everything is set up, simply toggle the "Use hardware" button in the upper right corner of Plugindoctor and all analysis signals will be routed through your hardware chain. From here on, all analysis will be performed in exactly the same way as if you were analysing an audio plugin. To stop analyzing your hardware and go back to plugin analysis, just toggle the "Use hardware" button again and select a plugin to analyze as usual.
Your hardware loop will inevitably have a non-zero latency, which will have an effect on the calculated phase response in the linear analysis section. This phase response is real, in the sense that the delay is actually taking place when you are integrating your outboard gear. Nevertheless, you might be interested in the latency-corrected phase response. For this purpose, click the "Adjust HW latency" button, wait one or two second, and the latency of your whole hardware chain will be taken into account. This needs to be redone when changing the sampling frequency.

Should any questions remain, or problems pop up during the usage of Plugindoctor, please do not hesitate to contact support@ddmf.eu!