# Parameters

**Enabled**

Controls whether the plugin is active. When set to “Disabled”, the effect is bypassed.

**Input Gain**

Controls the gain of the signal that is fed to the envelope detector. It does not affect the output signal, it only controls the sensitivity of the envelope detector. If your Open Threshold is overly sensitive, reduce the input gain. If you have set the threshold to the lowest possible value, but the signal is still not getting through, increase the Input Gain.

**Output Gain**

Controls the output volume of the effect.

**Attack**

Controls how quickly the signal fades in when the gate opens. Set to the lowest value possible that does not introduce pops or noise.

**Release**

Controls how quickly the signal fades out when the gate closes. Set to a low value for a sharp, dry sound. Set to moderate or high level for a less processed, “wetter” sound.

**Signal Floor**

Sets the gain when the gate is completely closed. Adjust so the amplifier output is very low or silent, but no lower. Setting this value too low can add a small amount of delay to the gate opening time, which is undesirable (reason for this is explained in the details description in the next chapter).

**Ratio Open**

The expansion ratio for the gate attack. Set to low values for a smoother fade-in. Set to higher values for a cleaner, faster attack.

**Ratio Close**

The expansion ratio for the gate release. Set to low values for a smoother fade-out. Set to higher values for a sharper cutoff.

**Threshold Open**

The signal level at which the gate turns on.

**Threshold Close**

The signal level at which the gate turns off. This value is negative and is relative to the Open Threshold, as the effective close threshold is Threshold Open + Threshold Close. By setting the close threshold lower than the open threshold, you can minimize “chatter”, where the signal rapidly cuts in and out when it is close to being cut off (gate closes).

**Knee**

Sets the sharpness of the expansion curve around the threshold point. Set to a lower value for a sharper change in the opening and closing of the gate. Set to a higher value for a smoother transition between the open and close states.

# The Algorithm

This noise gate contains 5 main parts:

* Signal Shaper
* Envelope Follower and Peak Detector
* Envelope Filter
* Expansion Curve
* Slew Limiter

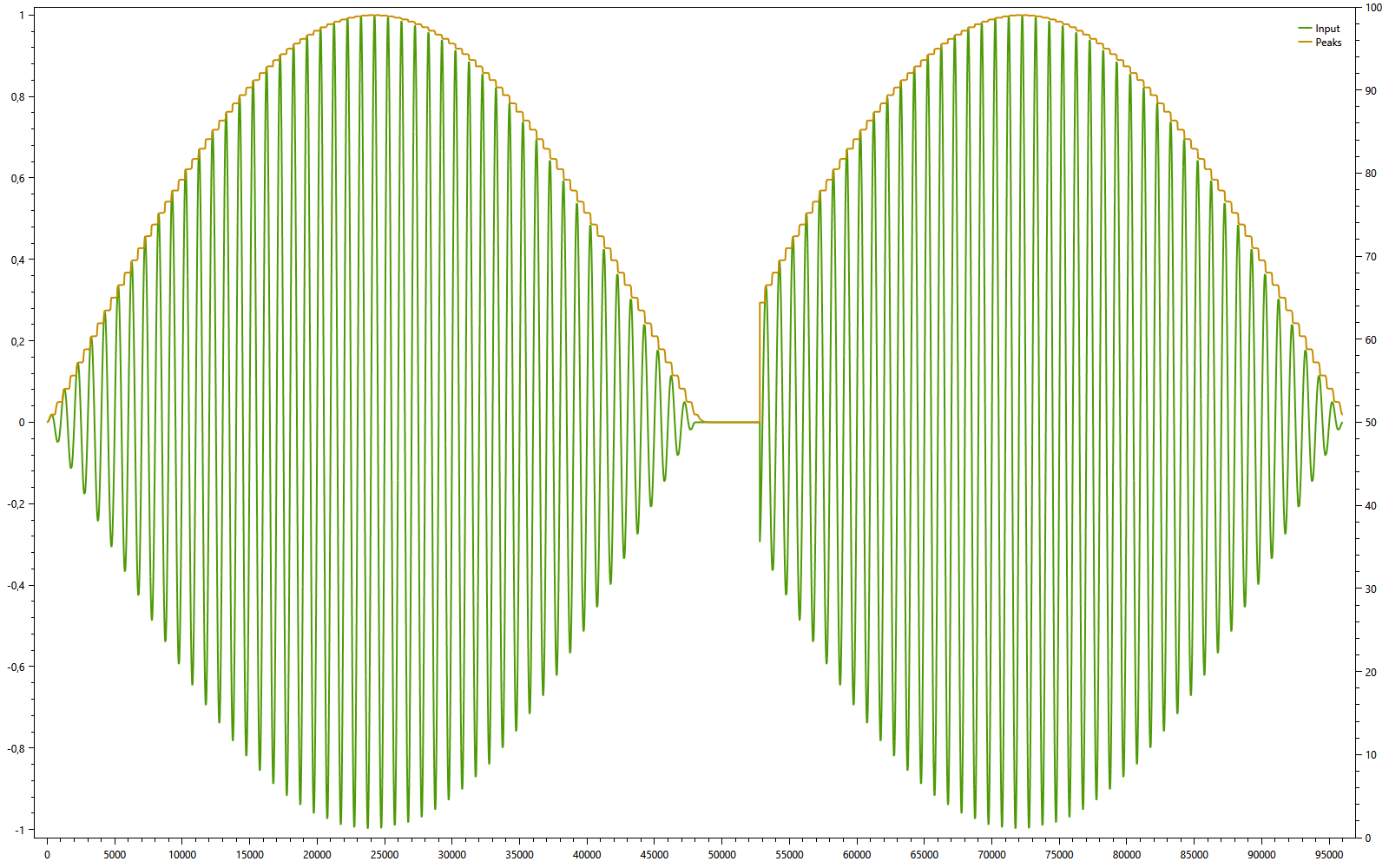
## Signal Shaper

The signal shaper is a simple 2 pole Highpass + Lowpass filter that is applied to the input signal before it hits the Envelope Follower. These values are fixed at 60Hz for the highpass filter and 4Khz for the lowpass. These filters will remove excessive DC offset from the signal (which may falsely trigger the noise gate to open), as well as noise in the higher range of the audible frequency spectrum. The pitch range of a standard guitar is 80Hz – 1300hz, so any signal outside that range does not help accurately detect the guitar sound.

## Envelope Follower and Peak Detector

The envelope follower’s job is to estimate the “signal strength” of the input signal, and output a continuous value that “envelopes” the signal. It does this in the following steps:

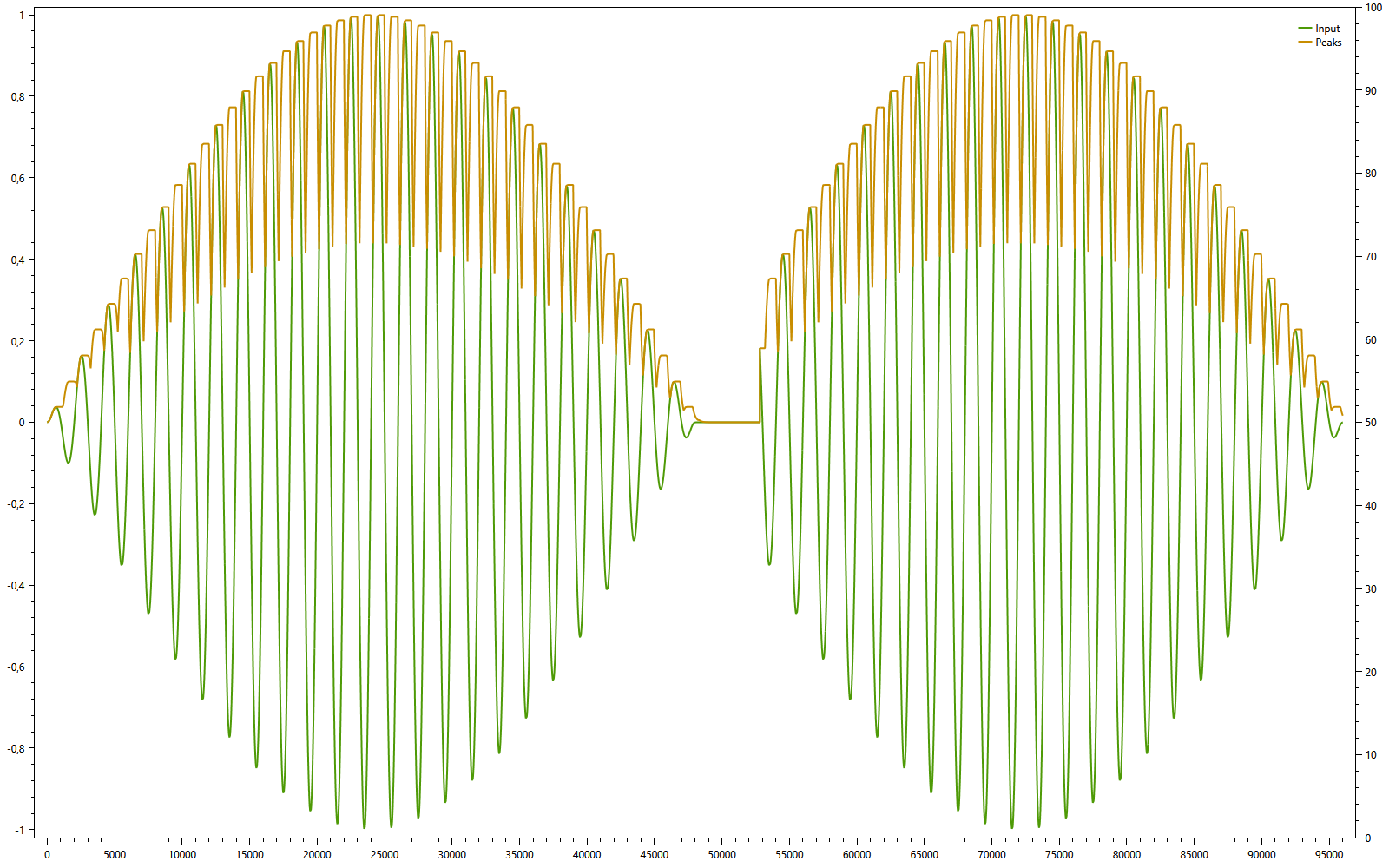
* Rectify the signal by taking the absolute value of the input sample.
* If the new value is greater than the one we saw previously, we store the previous sample as a “peak value”, and add it to an array of such values.
* A “hold” is applied to each peak value, for a fixed amount of time of 10ms. This helps smooth the between the peaks of the signal.
  + During each iteration of the process, we look back at the last 10ms of audio, and choose the largest peak signal from within that period.
  + If no peak has been detected in the last 10ms, the signal rapidly decays to zero.



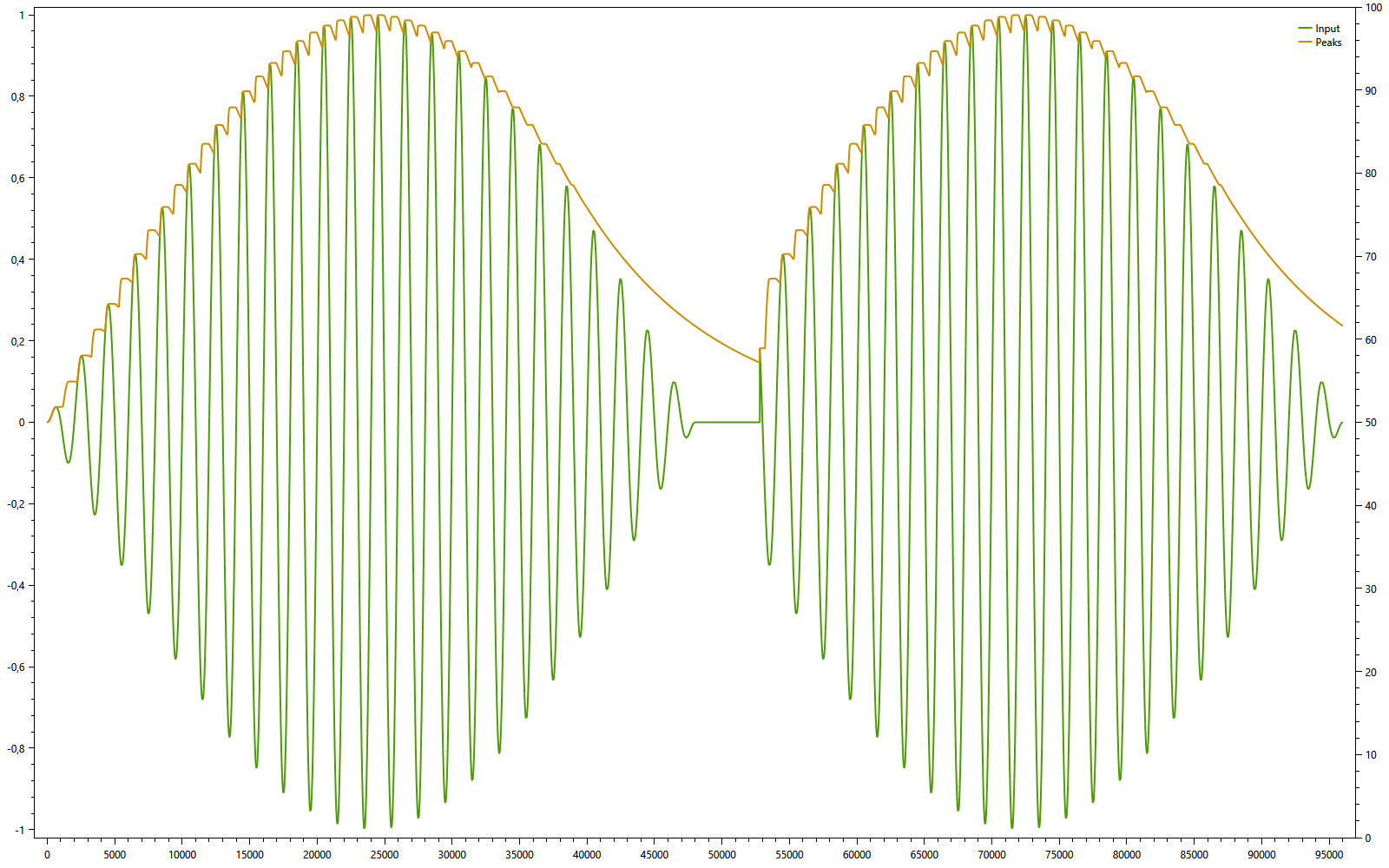
Envelope detector working as expected.



A signal with a very quick attack and decay. Notice how the envelope follower lags slightly when the signal cuts off, this is because the decay back to zero is not instant, but gradual.



A very low frequency signal. Here the envelope follower’s hold time is not long enough to smooth between the peaks; the envelope curve is not representative of the actual signal, and contains lots of “valleys”.



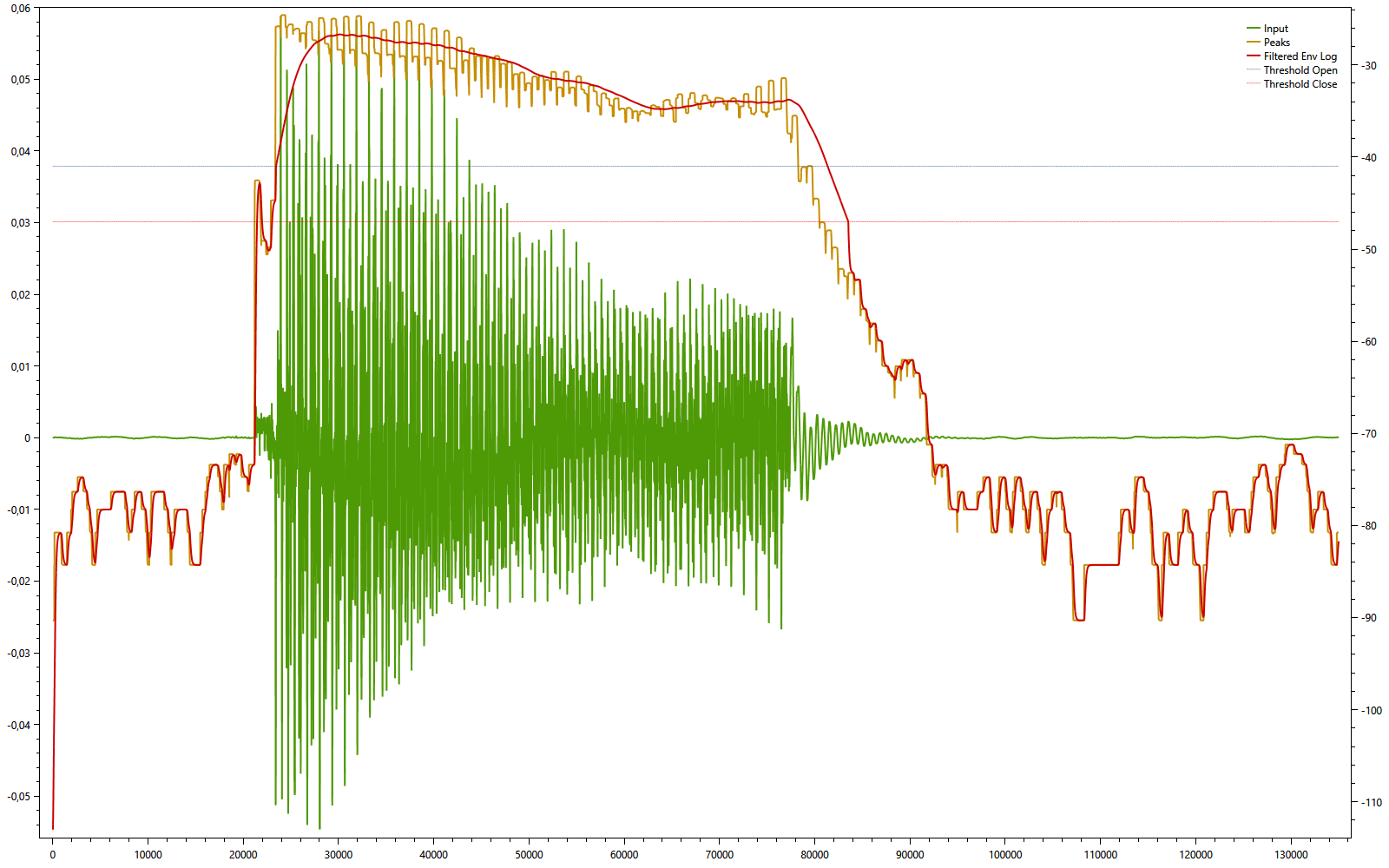
We can compensate by lowering the decay rate of the envelope follower. This gives a smoother signal, but it takes longer to decay to zero when input signal turns off. A decay value of 0.995, which is fixed in code, seems to provide a very good trade-off between the two extremes for guitar sounds.

## Envelope Filter

The signal from the envelope follower can be further smoothed, to obtain a better result. Two simple, 1 pole low pass filters are applied to the envelope.

Rather than using a static cutoff frequency, the cutoff is varied depending on whether the envelope is above or below the thresholds. The reason for doing so is that we want to quickly detect when the signal rises above the threshold, so a faster-changing signal is preferred (higher cutoff frequency). When the signal is above the threshold, we want to keep it smooth and steady for as long as it remains above it, so a slower-changing signal is preferred (lower cutoff frequency).

* When the envelope is above the Open Threshold, a cutoff of 10hz is applied.
* When the envelope is below the Close Threshold, a cutoff of 100hz is applied.
* When the envelope is between the Open and Close Thresholds, the cutoff remains unchanged from what it was before (hysteresis).



An example of how the smoothing is applied. Notice how the signal is considerably smoother above the open threshold, and becomes noisier when it falls back below the close threshold.

(Note that this chart has two y-axes, the input signal is on the linear axis on the left side and the other signals are decibel values on the right side axis).

## Expansion Curve

An expander is the opposite of a compressor. Instead of limiting peaks in volume, it reduces the volume of weak signals. This is controlled by an “expansion curve”, which acts as a mathematical translation. Given a decibel value, it translates it to a new value. The function has 3 parameters, in addition to the input value:

* Threshold – The “cutoff point”, where any value above the threshold remains unaffected, and values below it are reduced by some factor.
* Ratio – The amount of reduction that gets applied. This changes the “slope” of the curve below the threshold.
* Knee – This smooths out the curve around the Threshold point. Instead of having a discontinuity in the curve, the Knee smooths the two parts of the curve so they blend together.

A picture says more than a thousand words. Let’s look at some examples:

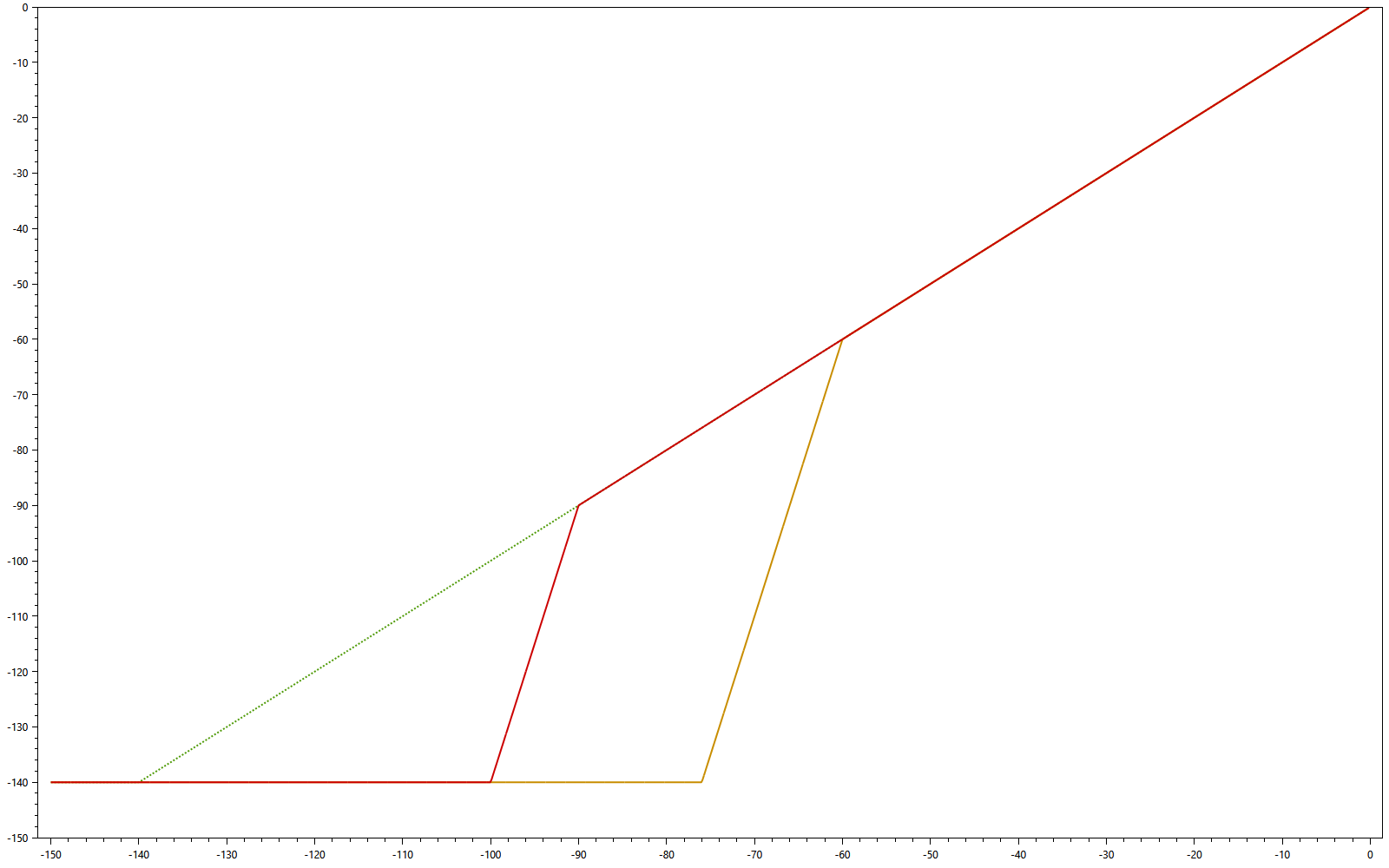
|  |  |
| --- | --- |
| C:\Users\Valdemar\AppData\Local\Microsoft\Windows\INetCache\Content.Word\chart2.png  Varying the Threshold | C:\Users\Valdemar\AppData\Local\Microsoft\Windows\INetCache\Content.Word\chart2.png  Varying the Ratio |
| Varying the Knee | |

The expander I use is different in two critical ways:

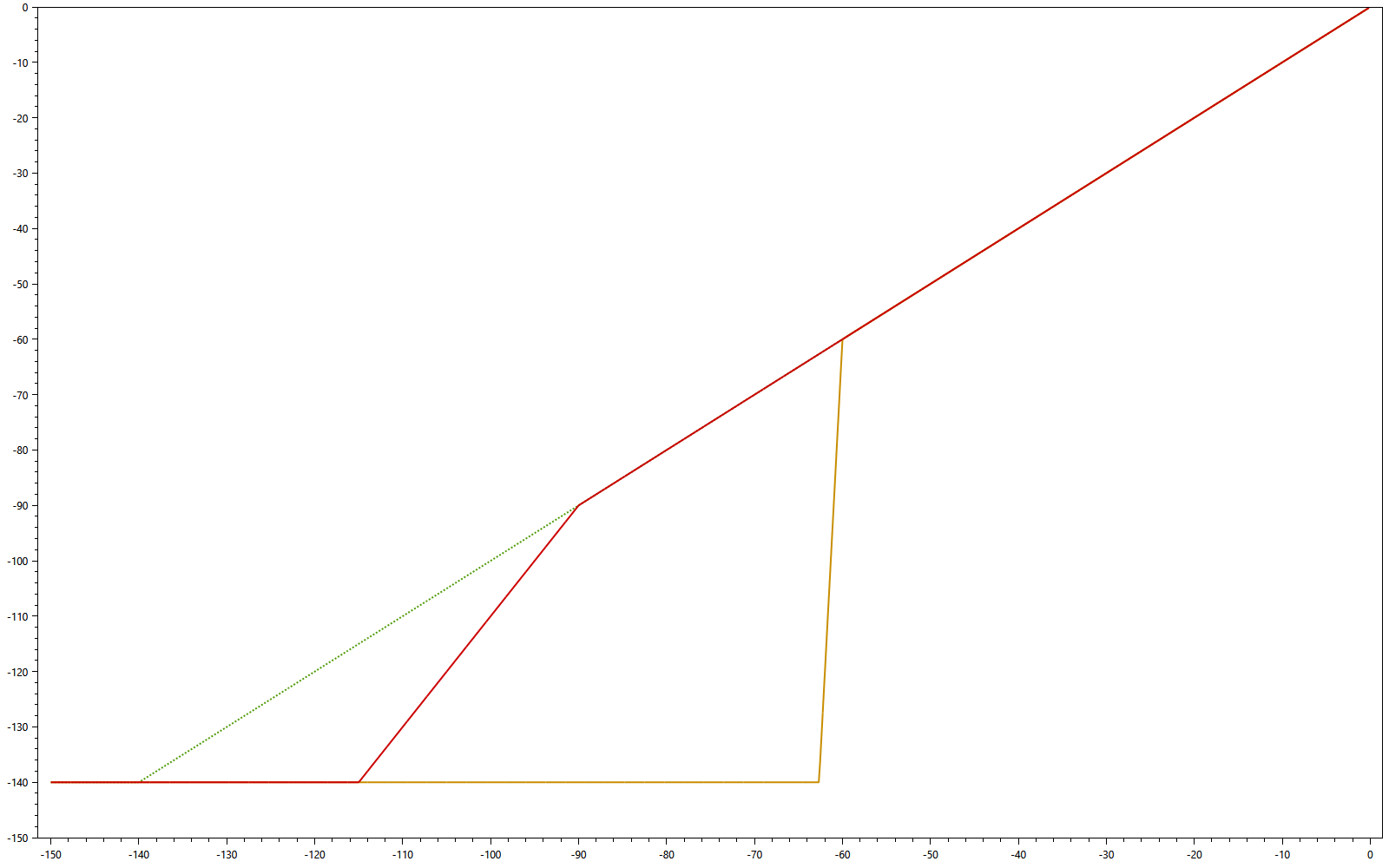
Firstly, it has a lower limit, set by the Signal Floor parameter. This restricts the expander from outputting a value lower than the stated signal floor.

Note that setting the signal floor too low means that, when the gate opens, the signal has to change from a very low value to the desired value. This change is not instantaneous, so the lower the noise floor, the slower the attack will react. Usually we want the attack to be as fast as possible, or it can cause a delay in audible sound, which is not desired. Setting the signal floor “just low enough” is the best approach.

Second, it actually uses two expansion curves, with different thresholds and optionally different ratios as well. The signal is “clamped” between the two curves, forcing it to follow different paths when increasing or decreasing. This has the effect of applying one expander to the signal attack, when the gate opens, and another to the signal release, when it closes.



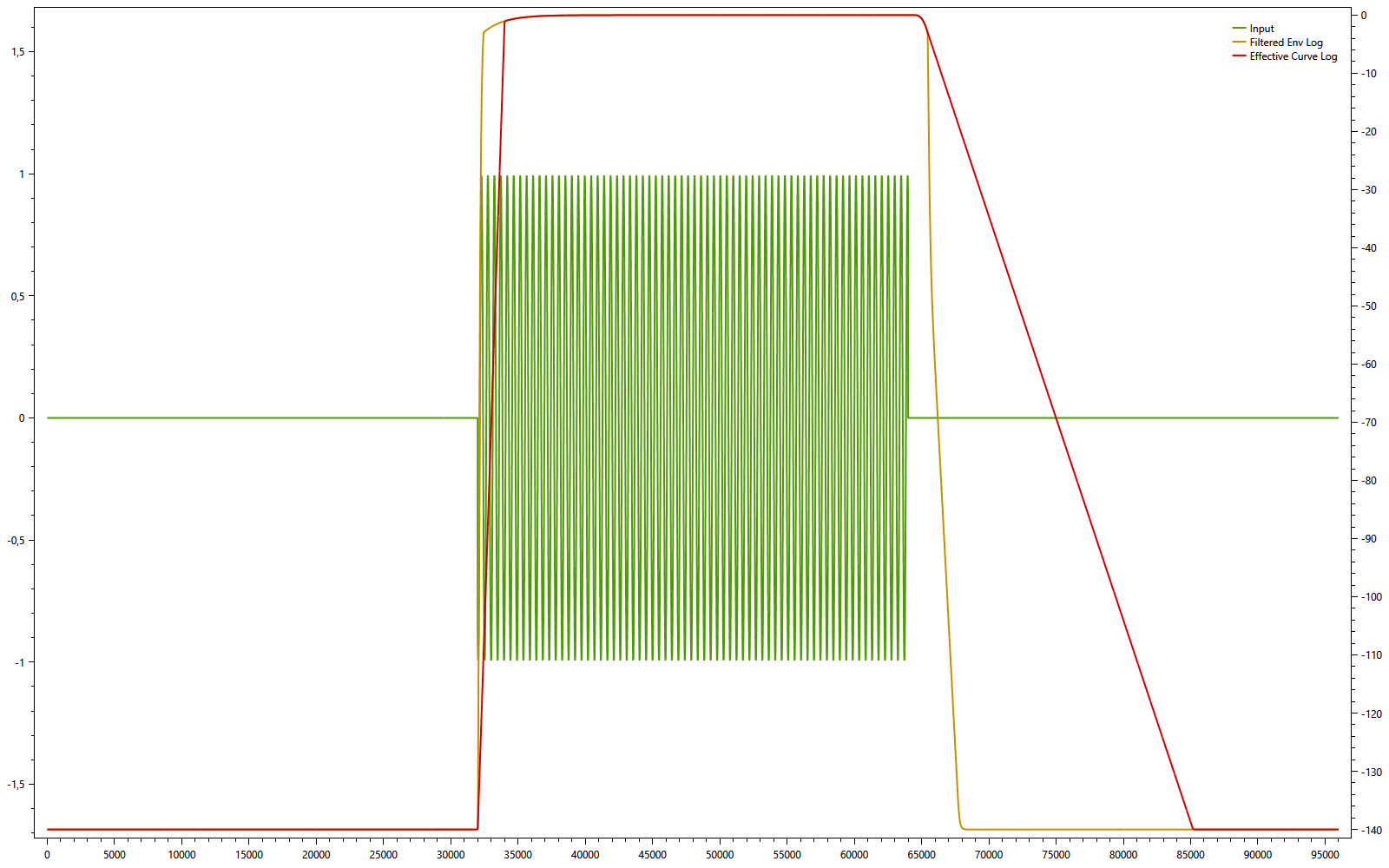
Two curves with different thresholds (-60 and -90). During the attack, the signal will be clamped above the yellow curve and follows that trajectory. During the release, the signal is clamped below the red curve, and follows that trajectory.



By using different ratios, we can achieve both a quick attack, and a smoothly decaying release.

## Slew Limiter

Finally, a slew limiter is added to the output of the expansion curve translation. A slew limiter simply limits how quickly a signal can increase or decrease. These parameters can be further used to smooth out the attack and release of the gate.



Yellow line: Unlimited signal.

Red line: Slew limited signal (30ms Attack, 300ms Release).