ProAudioDeclipper User Manual (v1.7.0)

Introduction

ProAudioDeclipper is a plugin developed by MyAudioFactory It is a powerful real-time audio processing tool that enhances audio quality by restoring the original dynamic range of clipped and saturated audio, effectively eliminating distortion caused by clipping and saturation. Unlike traditional declipper algorithms, ProAudioDeclipper uses an Innovative Algorithm to restore the original signal from clipped data. In many cases, the result is indistinguishable from the original, unclipped recording.

ProAudioDeclipper is an automatic audio de-clipping tool that requires minimal interaction. It can automatically detect clipping and saturation artifacts and restore them. Additionally, it features a soft peak limiter to adjust the final output level.

The flowchart for ProAudioDeclipper is as follows:

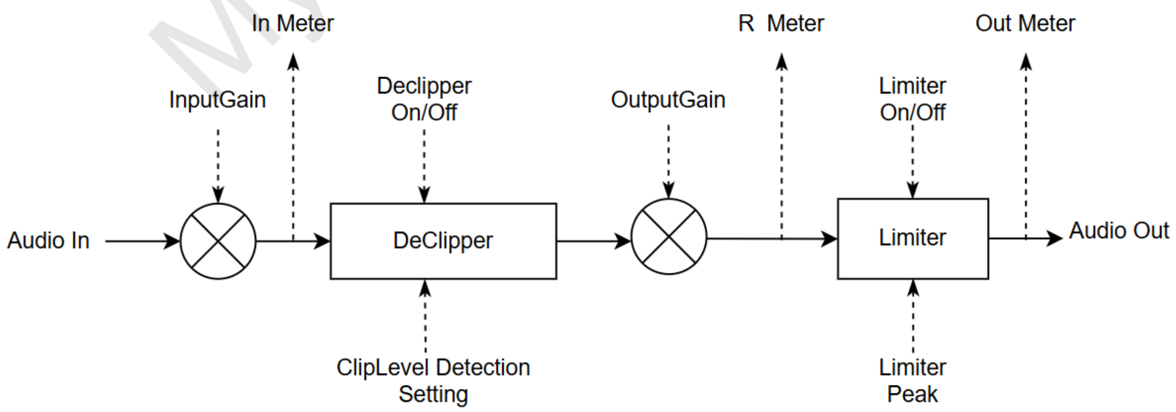

FIGURE 1

How to Install

For Windows, unzip the downloaded ProAudioDeclipper_Plugins_Installer.zip file, right-click the vst3_installer.bat or vst_installer.bat script, and run it as an administrator.

For macOS, run the downloaded ProAudioDeclipper_Plugins_Installer.pkg and follow the prompts in the Apple installer to complete the installation.

After installing the plugins, most Digital Audio Workstations (DAWs) can find the installed plugins automatically after being reopened. If not, you may need to manually rescan for plugins from the DAW's options.

How to register with Activation Code

Copy your activation code to the text field located in the license window, and then click "Confirm". If you would like to take advantage of a free trial, start ProAudioDeclipper and click the "FreeTrial" button in the license window.

“FreeTrial” is limited to limit days with limited features, and the total duration for processed audio files is limited to 1200 seconds. The processed audio has random 1-second intervals of silence every 20 seconds. Some settings options are locked, so the restoration capabilities during the free trial may not be fully optimized.

How to Transfer an Online License to Use on Different PCs

Once you activate the plugin with an online license activation code, a "Deactivate" button will appear on the license window. If you want to use this plugin on other PCs, please click the "Deactivate" button on this PC to release the license. Then, you can use your online license activation code to activate it on other PCs.

How to register with License File

The License File is designed to activate the plugin on a PC without an internet connection.

- Click the ‘Request’ button in license window, save the request file, and send it to us. We will send back a License File based on the request file.
- Click the ‘License File’ button in license window to select the License File for this host, then click the ‘Apply’ button to activate the plugin.

How to change a Setting value

The sliders can be adjusted by dragging it or using the mouse wheel. For more precise value changes, hold the SHIFT key while adjusting. Additionally, you can edit the value directly by double-clicking the value field.

When this plugin operates in real-time mode, changes to settings take effect immediately. However, adjusting the “SpeedVsQuality” or “Look-ahead samples” settings during playback may result in a short-term audible artifact, known as a 'spark.' It is recommended to make these adjustments when the plugin is not in playback mode, or when it is in a stopped status. The 'Real-Time Mode' indicator on the user interface confirms that the plugin is operating in real-time mode.

Settings:

General you do not need to change the settings, as the default configuration can handle most cases. However, if you wish to enhance performance beyond the default settings, please follow the introductions below.

Declipper Settings

1 Declipper On/Off Switch

Enable or Bypass de-clipping processing. When bypass de-clipping processing, input gain, output gain and Limiter can still work.

2 InputGain

Input gain of the audio sample before de-clipping processing. Input audio samples with a relatively large headroom can make it difficult to detect the audio samples' clipping level. When **“DynamicThreshold”** parameter is enabled, the algorithm will detect the clipping level automatically and we recommend adjusting the input audio data so that it has a clipping level greater than -12 dB (ideally at least -20 dB) for improved clip level detection.

3 OutputGain

Output gain of the audio sample after de-clipping processing.

4 SpeedVsQuality

It is a balance parameter between de-clipping processing speed and de-clipping processing performance. High value has better performance, low value has better processing speed. In most cases, value 3 or 4 is good enough.

5 Processors

The number of CPU processors that can be utilized for de-clipping processing directly affects the speed. Using multiple processors can significantly speed up the de-clipping process. For detailed instructions on how to adjust this parameter, please refer to **Tips about Settings and Usage** and **Running Mode:**

6 Look-ahead samples

It is used to better restore the original samples. More look-ahead samples result in better restoration performance, but require more computational power and latency.

7 Fix glitch On/Off Switch

Fix significant audio glitch or not.

8 LimiterPeak

Limiter Peak parameter to determine the peak value of output audio samples when limiter is enabled.

9 Limiter On/Off Switch

Apply a low distortion Limiter for final audio samples output or not. The peak value of the limiter is determined by the Limiter Peak parameter. When the limiter is turned off, it is bypassed.

Declipper Detection Settings

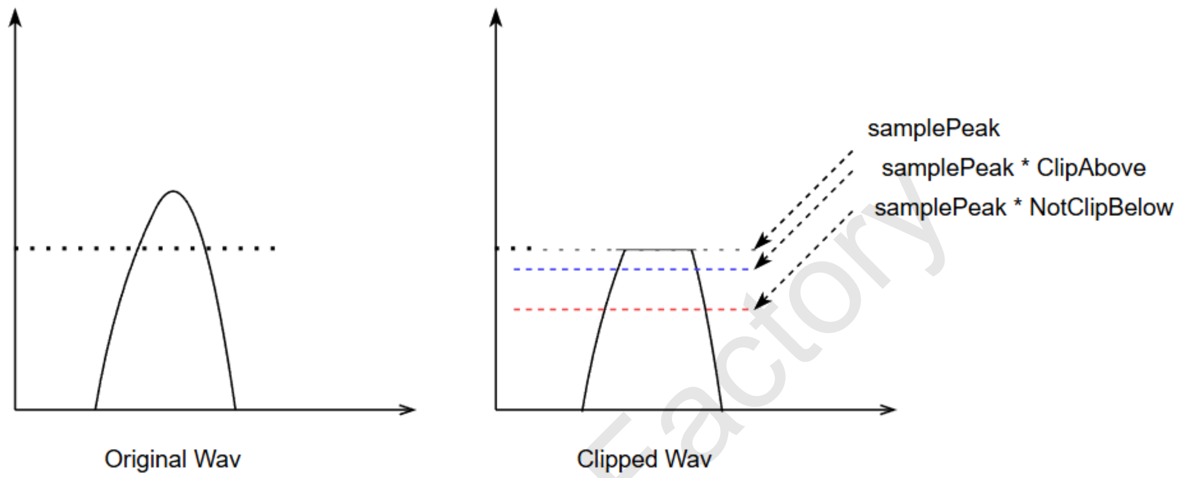

FIGURE 2

1 "Dynamic Threshold"

In Figure 2, the "samplePeak" represents the peak sample value of the clipped waveform. **"ClipAbove"** determines the blue line; all samples greater than this blue line will be treated as clipped samples. **"NotClipBelow"** determines the red line; all samples lower than this red line will be treated as normal samples.

If the **"DynamicThreshold"** parameter is disabled, samples greater than the red line will be treated as clipped samples. These clipped samples will be reconstructed by de-clipping processing.

If the **"DynamicThreshold"** parameter is enabled, the algorithm will find an optimal level between the red line and the blue line as the clip level. If samples exceed this clip level, they are treated as clipped samples. These clipped samples will be reconstructed by de-clipping processing.

The internal restriction ensures that the red line is always lower than the blue line in Figure 2.

2 "ClipAbove"

Typical value is 98% to 100%(0.98~1.0). It's a scale value to determine the high threshold of clip level. See the blue line in Figure 2. Audio samples larger than (Sample Peak * ClipAbove) are treated as clipped samples. These samples will be restored by de-clipping processing.

3 NotClipBelow

Typical Value is 99% to 25%(0.99 ~ 0.25). It's a scale value to determine the low threshold of clip level. See the red line in Figure 2. Audio samples smaller than (Sample Peak * NotClipBelow) are treated as unclipped samples. De-clipping processing will not touch these samples.

4 NotClipAmp

Audio samples smaller than NotClipAmp will never be considered clipped samples. The De-clipping processing will not touch these samples. This is a parameter to prevent user set a very low value of NotClipBelow. If the Red Line in Figure 2 is lower than this "NotClipAmp", it will be restricted to "NotClipAmp".

5 PeakDisRate

The sample peak is the maximum value of an audio sample over a period of time. Given some anomalous samples, it is reasonable to discard some of the largest samples as sample peaks. Generally, a discard rate of 1 in 10,000 is reasonable.

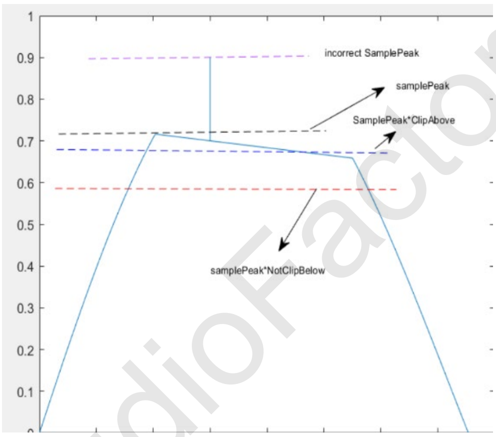

FIGURE 3

In some cases, such as when clipped audio has been transcoded by a lossy codec, or when audio is extracted from damaged CDs, etc., there are some anomalous samples present in the clipped audio, which can affect the algorithm's ability to find the correct sample peak, as shown in Figure 3. The '**PeakDisRate**' parameter controls the algorithm to discard some maximum values within the search window in order to find the correct peak sample.

Another trick to achieve better results is: Increase the “**InputGain**” parameter to make the actual sample peak approach 0dBFS. Then, Figure 3 will resemble Figure 4, and the declipper will find the correct sample peak.

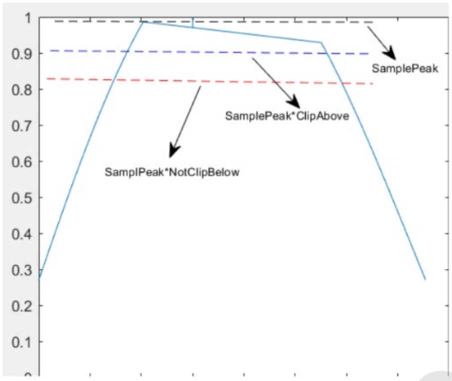

FIGURE 4

6 PeakTime

Length of time window to find the Sample Peak.

7 ResetAmp

For streaming audio, different audio clips usually use muted or low amplitude audio as transitions. Different audio clips may have significantly different clip levels. ResetAmp is used as an amplitude threshold for detecting transition audio. Audio smaller than ResetAmp may be transition audio between different audio clips such as Figure 5. Clip 1 and Clip 2 are different audio clips that may have significantly different clip levels. ResetAmp and ResetTime are used to detect these audio changes and reset some internal states to achieve more accurate detection of audio clip levels for the early portion of audio clip 2.

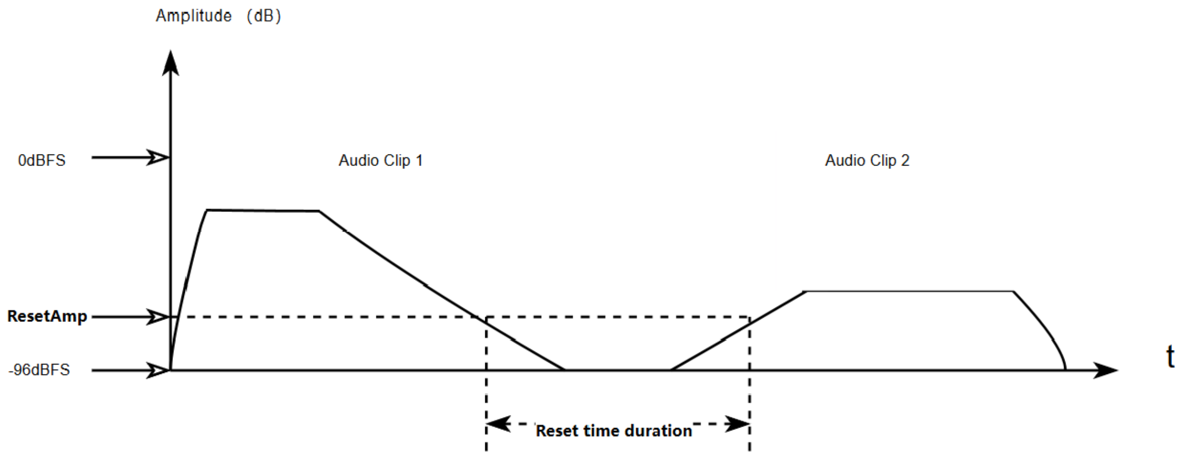

FIGURE 5

8 ResetTime

The minimum time for transitions between different audio clips is determined by the ResetTime. If the amplitude of the streaming audio remains consistently below the ResetAmp threshold throughout the ResetTime duration, the detector will consider the subsequent audio as a new audio clip. In other words, if ResetTime less than the rest time duration as shown in Figure 5, Audio Clip 2 will be treated as a new audio clip.

ResetAmp and ResetTime can be utilized when you intend to perform batch de-clipping processing on various audio clips.

Meters of Declipper

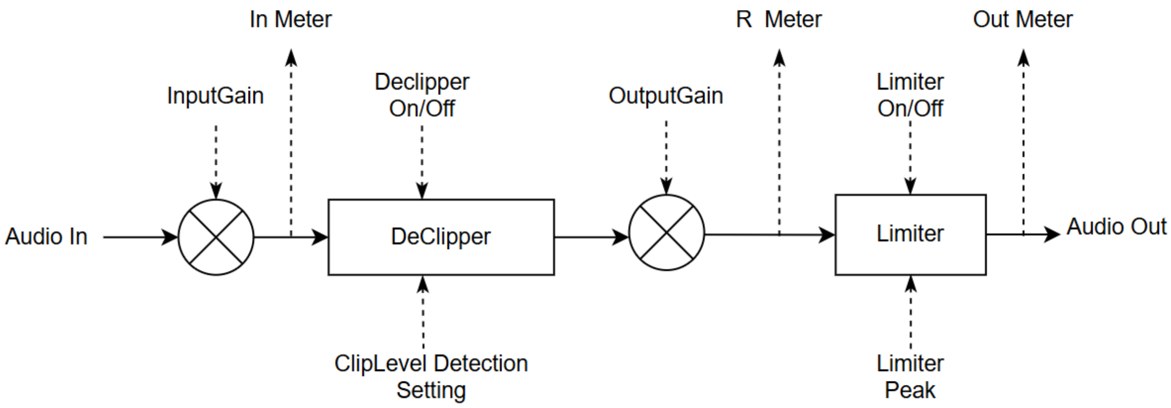

FIGURE 6

There are three meters used to indicate the audio amplitude at different stages:

1. The Input meter displays the audio's amplitude with input gain applied. It assists in adjusting the Input Gain so that the audio's peak level is above -12dBFS to facilitate clip level detection automatically.

   Yellow in this meter indicates the amplitude that is considered clipped. It can assist in visualizing the clipped level, which is useful when you wish to adjust the clip detection parameters for improved results.

2. The Restored Audio meter indicates the audio levels after output gain has been applied. If the meter shows levels in red, it signifies that the restored audio samples have exceeded the Limiter's Peak threshold. To avoid unintended dynamic range compression at the limiter, you should consider reducing the

output gain, particularly if the meter indicates levels above the Limiter's Peak by more than 3dB, unless this is the intended effect.

3. The Output meter shows the final audio amplitude.

4. Plugin running mode: ‘RealTime Mode’ and ‘Offline Mode’. This status Indicate whether the DAW is running this plugin in real-time or offline mode.

Tips about Settings and Usage

1. If you run this plugin in real-time mode, it's important to adjust the audio buffer size to prevent audio glitches. Typically, an audio buffer size larger than 512 samples should suffice. The declipping process is computationally intensive and may lead to the output audio buffer underrunning if your computer's processing power is not sufficient, especially when the "SpeedVsQuality" parameter is set to a high value or "Look-ahead samples" parameter is set to a high value.

   To optimize performance, you can adjust the processor parameter to find a suitable value that maximizes the utilization of your CPU's computational capability.

2. If you run this plugin in offline mode, to achieve the best restoration performance, please set the "SpeedVsQuality" parameter to its maximum value and set the "Look-ahead samples" parameter to its maximum value as well. Additionally, to speed up the rendering process using all available CPU processors, set the "Processor" parameter to its maximum value.

3. When dealing with clipped audio that is at or near 0dBFS, to provide the necessary headroom for restoring the clipped audio, you can either attenuate the input audio beforehand or set a negative post-gain in the settings.

4. If you already know the specific clip level you prefer for restoring clipped audio, please set the appropriate 'ClipAbove' parameter, which determines your preferred clipping level. Then, set the 'NotClipBelow' parameter to its maximum value.

5. For clips that have been transcoded using lossy codecs such as MP3, AAC, OGG, etc., you may slightly increase ‘PeakDisRate’ parameter. This adjustment allows the clipping detector to find a more accurate clipping level.

Running Mode:

1. Real-Time Mode

   DAWs can run effect plugins in real-time mode, allowing users to hear the plugins' effects in real time.

   In real-time audio processing, to maintain real-time performance and low latency, DAWs call plugins to process audio in short blocks, such as every 128 samples. This approach can limit the benefits of multi-processor utilization. For instance, if a plugin requires a minimum processing block of 128 samples for tasks like a 128-sample FFT, and it receives exactly 128 samples, it will be processed by a single processor since that is the minimum audio block size. Consequently, even if your CPU has multiple physical core processors, they cannot be fully utilized in this scenario.

In other words, how many effect plugins can run smoothly in real-time mainly depends on the singled CPU core's capacity.

Even though the ProAudioDeclipper plugin is capable of running on multiple processors, it may not gain much benefit when the DAW runs it in real-time mode. It may even reduce the CPU's compute performance if you set the "Processor" parameter to its maximum value in real-time mode, especially when the DAW has many real-time plugins to run, as this can cause the CPU to become busy with scheduling tasks on limited physical resources. We recommend setting the "Processor" parameter to a value between 2 and 8, depending on your actual CPU capabilities.

In real-time audio processing, it’s critical to adjust the audio buffer size in a DAW to prevent audio glitches. A small audio buffer size in a DAW will increase CPU overhead and reduce the CPU’s real computing power. Typically, an audio buffer size larger than 512 samples should suffice.

2. Offline-Mode

In offline mode, a DAW typically processes audio by calling plugins with large data sizes, handling it in blocks of 1024 or even 8192 samples. If a plugin is capable of multi-processor operation, it can divide these blocks into smaller segments—for instance, splitting 1024 samples into 8 pieces of 128 samples each—and then distribute these segments across different core processors for parallel processing. This approach can lead to a significant performance improvement when using multi-core processors.

Since the ProAudioDeclipper plugin has the capability of running on multiple processors, we recommend setting the "Processor" parameter to its maximum value to gain the full benefit of multi-processor use when the DAW runs it in offline mode.

Known Issue:

N/A

Support info:

Please send email to contact@myaudiofactory.com or leave message via our website: https://www.myaudiofactory.com/