IDENTIFYING AND ANALYZING RELEVANT CHARACTERISTICS OF DYNAMIC RANGE COMPRESSION

Andrés Cabrera

Departamento de Música
Pontificia Universidad Javeriana, Bogotá, Colombia
andres@geminiflux.com

ABSTRACT

This document presents a method for measuring, identifying and analyzing the fundamental characteristics of dynamic range compressors. Emphasis is placed on making the analysis and testing method as automated as possible. Several characteristics of dynamic range compression have been identified by comparing the behavior of six different plug-in compressors.

1. INTRODUCTION

Dynamic range compression has always been an important aspect of audio processing both for aesthetic and technical reasons. Compression is basically an algorithm to derive a gain factor from an envelope follower, and there are many hardware devices that can achieve dynamic compression. With the advent of DAWs, many software plug-ins have been developed for this purpose.

Most compressors have four parameters in common: *Threshold, Ratio, Attack* and *Decay time*. Others may offer control of *Knee* and *Auto Release* control, and others offer different modes of operation (*Opto* or *Electro* in the case of the Waves RCompressor [1], *Vintage* or *Normal* in the case of the Sonitus Compressor [2] and selection between *Peak* and *RMS* tracking in the case of the SC4 [3]). Some have fixed Attack and Decay times, and some have a fixed threshold but have input and output gain control to adjust compression amount. Other parameters which exist in compressors but which act independently from the actual compression process like makeup gain, and side chain EQ will not be dealt with here.

The *Threshold* of a compressor is the amplitude above which gain reduction begins to be applied. Everything below this threshold should pass unaffected, but as will be seen, the *Knee* and the way the amplitude of a signal is tracked by the compressor affects dramatically the exact location of the threshold.

The *Ratio* of a compressor determines how much gain reduction should be applied to signals crossing the threshold. It is a ratio between the amount of original signal above the threshold and the amount of signal above the threshold after gain reduction.

The *Attack* and *Decay times* determine how long it takes for the compressor's gain reduction to fully kick in after a signal has crossed the *threshold*, and how long it takes for the compressor to stop reducing gain after a signal is once again below the *threshold*. These stages of attack and decay are referred to here as gain reduction and gain recovery.

A compressor's *Knee* affects the behavior of gain reduction around the threshold. A soft knee or knee with high radius results in compression being introduced mildly for amplitudes below the threshold. As the signal level goes beyond the threshold, the ratio of compression is increased until the target ratio is reached somewhere above the threshold. A hard knee introduces compression only on the threshold and only at the set ratio.

Even though all compressors process the signal in similar manner, the perceived and technical quality of a compressor's output vary dramatically.

This article will present an analysis method and some preliminary results for 6 different compressors: SC1 [4], SC4 [3], Tap dynamics [5], mda dynamics [6], Sonitus fx:compressor [2] and Waves Renaissance Compressor [1] attempting to identify the relevant characteristics of a compressor which determine the quality of its output. These compressors span most of the spectrum available in plug-in compression, including high-end commercial, freeware and open source plug-ins.

Although dynamic range compression of audio signals is almost as old as audio recording, no references could be found documenting similar testing or analysis methods, although certainly some must have been developed by the private industry for the purpose of emulation and design.

2. ANALYSIS METHOD

The principal effect of compression on an audio signal is that of gain reduction. Previous efforts to measure and model dynamic range compressors [7] have focused mainly on the processor's frequency response and the total gain reduction applied after a steady gain reduction has been reached. The primary goal of this particular project was to model vintage analog compressors whose frequency response at different gain reductions is particularly characteristic of the processor to be modeled. However, relevant characteristics of a compressor like exact treatment of transients and peaks, the actual gain curves, the effect of the input signal's frequency spectrum on the gain curves and the effects of the time variables are not considered. Other research has produced companding models designed as hardware replacement which though suitable for technical applications, do not seek to emulate desirable auditory characteristics of particular compressors [8]. There exist commercial plug-ins that model actual hardware, like the URS plugins [9], but these models are as closed and unknown as their analog originals and there seems to exist no documentation about the technical processes involved. Additionally the method presented here seeks to automate as much of the procedure as possible to optimize evaluation.

The analysis method proposed here seeks to analyze a compressor's characteristics in detail to cover all aspects of its operation. The method makes the following assumptions:

1. The gain curve and reduction depend only on the amount

of signal above the threshold, and not on the actual threshold level. This means that a signal going 2dB above the threshold will be compressed always the same amount, independently of the actual threshold level.

2. The processor only affects the amplitude of a signal. All phase characteristics of a signal are retained.
3. A compressor can be fully described by its effect on single frequencies.

The first assumption will probably hold true for digital compressors since making the behavior of the compressor be dependent of the threshold level would increase the algorithm's complexity without much gain other than irregular behavior of the process at different levels. For analog hardware compressors this assumption may not be so correct since anomalies in analog circuitry might lead to irregular behavior. However the analysis method proposed here can be repeated for different threshold levels to build one complex algorithm or several different models.
The second assumption, as the first one, probably holds true for most digital compressors. The method does a crude check for zero crossing incoherences (which would occur if the process introduced phase irregularities), and none of the results, even in the case of white noise reported any discrepancies.
The validity of the last assumption will have to evaluated once the method has yielded further results. If it is seen this assumption leads to inconsistent results, the composition of the test sample (see section 1) will have to be adjusted to include more complex spectra than pure sine waves.

2.1. Characteristics of a compressor to be measured

This method proposes analyzing the following characteristics of a compressor:

1. The final gain reduction applied to a steady signal (constant sine wave or noise) after a constant gain reduction is reached. This gain reduction shall be measured for values well above and below the threshold, with particular detail around the threshold.
2. The effect of frequency on gain reduction applied, i.e. measuring differences in gain reduction for steady signals with equal peaks but different spectral content.
3. The effect of the signal's frequency on the shape of the gain reduction/recover curve.
4. The time required to achieve a steady gain reduction. This time will be compared to the actual attack time setting, and will be measured with different attack time values.
5. The shape, latency (see section 2.4) and duration of the gain reduction curve. This will reveal the exact treatment of peaks and transients in the compressor. This will provide additional clues as to the process the compressor uses to track amplitude.
6. The time required by the compressor to remove gain reduction when a signal's level decreases. This will be compared to the actual release time and will be measured with different release values.
7. The shape and time of the compressor's gain recover curve, i.e. the section where gain is restored after a signal goes below or comes down closer to the threshold.
8. The effect of gain reduction on the spectrum of broadband noise. This will reveal any coloring (EQ) applied by the compressor when reducing gain.
9. The shape of the compressor's knee, i.e. the way the gain reduction ratio varies around the threshold (going from 1:1 to the final ratio).
10. The characteristics and presence of non-linear distortion applied by the compressor and the effect of a signal's frequency on this distortion.

It is estimated that these characteristics accurately describe the behavior of a compressor, and will provide all information necessary for later modeling and analysis.

2.2. Test sample

The first step of the method consists in passing a test sample (in this test the sample rate used was 44100 kHz with 24-bit integer resolution) through the compressor. The compressor is used as a plug-in in any suitable host, and the resulting audio is exported either at 24-bit integer (for the Windows plug-ins) or 32-bit floating-point (on Linux, since the Linux host used permits this).
The current test sample (which is certain to continue evolving and was prepared using a Python [10] script to generate a Csound [11] score) contains the following parts:

Parts	Amp. (dB$_{FS}$)	Freq. (Hz)
0-7	0	50, 200, 800, 1500, 3000, 8000, WN, DC
8-151	-5 to -30	50, 200, 800, 1500, 3000, 8000, WN, DC
152-226	-5 ramp to -10 to -35	200, 1500, 8000, WN, DC
227-301	-10 to -30 ramp to -5	200, 1500, 8000, WN, DC
302-376	-5 to -28 jump to -30	200, 1500, 8000, WN, DC
377-451	-5 to -28 jump to -40	200, 1500, 8000, WN, DC
452-471	0, -10, -15, -20, -25	Square wave @ 50, 200, 1500, 8000
472-491	0, -10, -15, -20, -25	Saw wave @ 50, 200, 1500, 8000
492-511	0, -10, -15, -20, -25	Triangle wave @ 50, 200, 1500, 8000

Table 1: *The parts of the test sample.*

Each of the parts in the test sample is a short sound followed by silence. The duration of the sound and the silence are long enough for the compressor to apply full gain reduction and achieve a steady state, and then to recover completely from gain reduction. The parts are referenced by a number which will later identify the segment.
The test sample was designed to be used on a compressor with the threshold set to -20dBFS, and has the best resolution around that amplitude. It consists of the following parts:

- Parts 0 to 151 are designed to measure the compression gain reduction curve and the final gain applied to signals at different frequencies (Characteristics 1,2,3,4 and 5). Many amplitudes and frequencies are used to judge both the effect of level and frequency on gain reduction. White Noise and a DC offset are also passed to obtain additional information.

- Parts 152 to 301 (Characteristic 10) are designed to measure the knee shape both for gain reduction and gain recovering. Perhaps the latter will be seen redundant after further study if is found to be consistently similar.
- Parts 302 to 451 will reveal the compressor's gain recover curve (Characteristics 6,7 and 8) by making the compressor apply gain reduction and then suddenly lowering the volume of the signal to evaluate the gain recovery stage. These parts and the previous ones could be useful in determining the release characteristics for compressors with options like Auto Release Control.
- Parts 452 to 511 are designed to test the compressor for non-linear distortion (Characteristic 11). These known - waves exhibit well known characteristics than can easily reveal any non-linear distortion applied. Non-linear distortion may be part of a compressor's process as shown in [12]
- All parts containing white noise will be used to evaluate the compressor's frequency response (Characteristic 9).

A single sample (of 20+ minutes length) was produced instead of many smaller ones, because it makes the processing through the compressor a one pass process, and it facilitates the redesign and modification of the test sample. This single sample will produce a new processed sample which will provide information on the behavior of the compressor at a particular setting. Several processed samples must be obtained at different settings to fully evaluate a compressor's behavior. Additionally the test sample has a pulse on its first sample to test and later compensate for any delay introduced by the compressor or the host.

For the current tests, all compressors were evaluated at their different modes when available (Opto-Electro; RMS-Peak; Vintage-Normal) and at all permutations of the following:

- Attack and release set equally to: 20ms, 40ms, 80ms and 400ms
- Ratios of 2:1 and 4:1

If a compressor had a knee setting, relevant settings were chosen, and all the measurements were done for them as well.

This is one the most time consuming processes of this method, since currently hosts cannot be scripted and ordered to change a plug-in's parameters between passes (on Windows), so this and the generation of the processed file must be done manually. This is not very problematic for simpler processes like the SC1 which might take half a minute to process (on a P4 1.6GHz machine), but it is definitely time consuming for CPU hungry plug-ins like the Sonitus which takes around 15 minutes to go through the sample on the same machine.

It is important to maintain consistent naming conventions throughout, especially since this analysis yields large numbers of files. In this case, the test file names included version number and threshold target, and the processed files included in their names information about the time and ratio settings as well as the name of the plug-in, the version and threshold target of the original test file and any other important settings used.

Notice that the amplitudes are calculated in absolute peak (dBFS) values. Since compressors usually track amplitude using some type of RMS or integration technique, discrepancies are expected in the gain curve results according to frequency. It may be necessary in future versions of the test sample to provide equal RMS amplitude values instead of equal peak values. However since the tracking method varies so dramatically between compressor implementations, using the absolute peak values is a valid compromise, and provides useful data.

2.3. Comparing the samples

To obtain the gain reduction curve, the resulting sample must be compared to the original test sample. This can be done mainly in two ways:

- Comparing amplitude values directly for each sample.
- Comparing RMS values, or integration.

The most straight forward way is the first one, which involves simply dividing the resultant sample by the original sample. It is important to have absolute frame accuracy otherwise this process will yield unpredictable results. It is also important to ignore the results produced by samples which are 0. In this method, if the samples are 0, the result for the previous sample is used, though interpolation might be a more accurate method. This is the way gain reduction has been evaluated so far using this method, though some tests comparing RMS values have also been done. The process has been realized using a Csound program (running at kr=1) which on a single pass computes the gain reduction curves and generates a result file for each segment of data (corresponding to the parts listed on section 1).

2.4. Characteristics identified

From the analysis performed on 6 compressors, the following characteristics which seem to have impact on a compressor's quality have been identified:

1. *Latency*: This is the time required for a compressor to start acting. This characteristic is possibly related to the way a plug-in measures level. For instance to calculate the RMS level of a signal, one point is not sufficient, several points must be evaluated. It is possible that some plug-ins do not compensate for this delay and generate *Latency* when applying gain reduction. This latency was observed on the free and open source plug-ins, but it was absent (even to the point that gain reduction was applied to the first sample in the case of the Waves Rcomp). Where observed, latency depends on the amount of signal above the threshold. The higher the signal above the threshold, the lower the latency (See figures 1 and 2).
2. *Smoothness*: The smoothness of the gain reduction curve seems to be another important factor in the way a compressor sounds. Rugged gain reduction curves might be the result of rounding up or irregularities in an algorithm. These irregularities seem to be present only on the SC1, SC4 and Tap plug-ins (Figure 3).
3. *Single frequency handling*: Refers to the way the compressor reacts to different frequencies. All frequencies (more noticeably lower ones), depending on the way the compressor measures level, might trick the compressor into thinking they are actually a fluctuating signal, when they are not, since the window for measuring the level might not cover

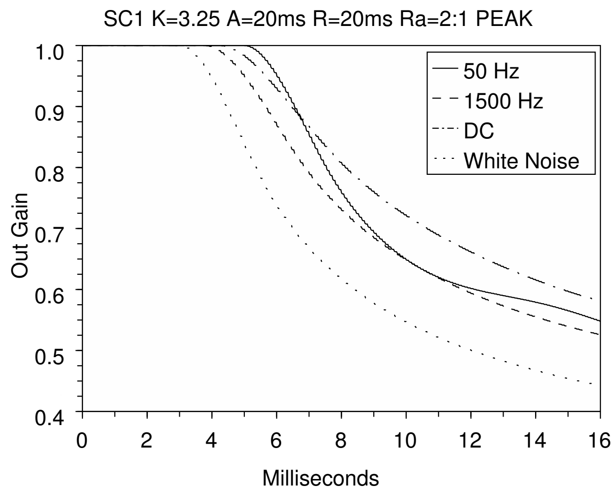

Figure 1: *Gain reduction (PEAK) for the SC1 plug-in at different frequencies.*

the whole wavelength of the signal. This artifact will certainly cause some degree of pumping or other undesired artifacts, and has been identified in all plug-ins tested (see the 50Hz curve in figures 1 and 4).

4. *Steady state frequency deviation*: When a steady gain reduction state has been achieved, plug-ins produce different amount of gain reduction for different frequencies. However, this doesn't amount to changes in frequency response, since the processed white noise doesn't exhibit any kind of coloring, but it will reflect how different kind of material (and especially with regards to spectral content), will be processed by the compressor. Some plug-ins exhibit very even response like the SC1, unlike the Sonitus plug-in which tends to apply less gain reduction to lower frequencies (Figure 4). All plug-ins tested exhibit very different characteristics, but it was observed that lower frequencies tended to be less attenuated, while DC offset tend-ed to be the most attenuated. This seems to point that all plug-ins tested use some form of RMS or integration method in their envelope follower.

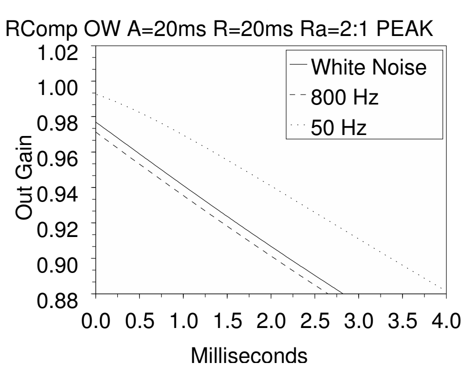

Figure 2: *Gain reduction detail (PEAK) for the RCompressor plug-in.*

2.5. Analysis of the gain reduction curves

A large number of files (1000+ for each result sample) are generated by the previous process. Using Scilab [13], a numerical computation program, it is possible to easily and efficiently extract information from the data, as well as provide graphical feedback which can guide the evaluation process.

It might also be desirable to calculate the derivative of the gain reduction curve. This has been done within Scilab using spline interpolation between each sample. This derivative will help identify when a steady gain has been reached, and may provide additional information about the gain reduction or recover curves.

A Scilab script has been written which can easily load and compare different sets of data from the files generated above. Information that can be extracted in this procedure include:

- Determination of the amplitude tracking technique. Figure 5 shows the RMS comparison value for the SC4 plug-in. The near coincidence of the derivate of the curve for all frequencies shows that this plug-in uses in fact an RMS algorithm. It is not shown here, but the deviations shown appear to be artifacts of interpolation and rounding.
- Localizing the point where gain has stabilized. This can be done again using the derivative of the function.

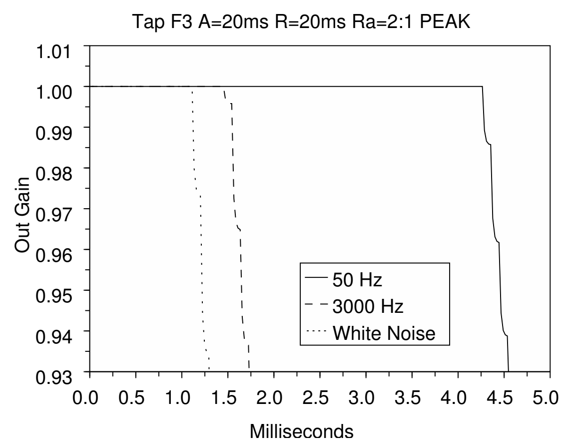

Figure 3: *Gain reduction (PEAK) for the TAP dynamics plug-in at different frequencies.*

- Determining the plug-in's *Latency*. In Figure 1, the SC1 exhibits a latency of around 4.5 ms. In Figure 6 the mda plug-in exhibits very different latencies for each frequency. The RCompressor has kicked in with gain reduction from the first sample (Figure 2).
- Determine the gain applied to a constant signal. Figure 4 shows how the Sonitus compressor behaves for signals 15 dB above the threshold. It can be compared to the behavior of the SC1, which exhibits a very different behavior (Figure 1). Observe particularly the difference in treatment of white noise, DC Offset and the 50Hz line.
- Compare the effect of amplitude above threshold and of the ratio on the gain reduction. See figure 7, which shows how the TAP dynamics plug-in applies gain reduction to signals of the same frequency but different amplitude. Notice the effect of signal level on latency.

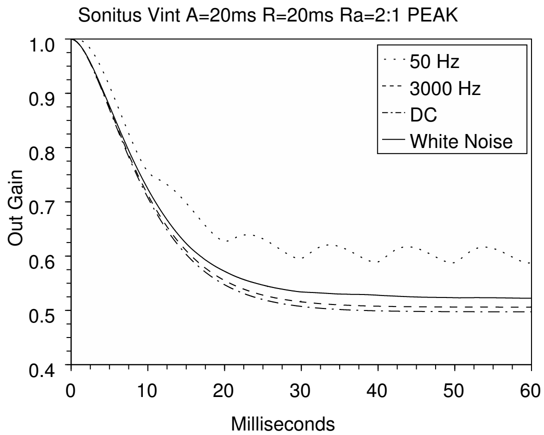

Figure 4: *Gain reduction (PEAK) for the Sonitus plug-in at different frequencies.*

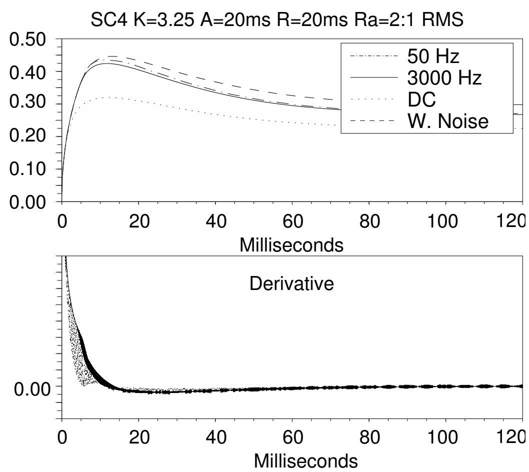

Figure 5: *RMS Gain reduction and its derivative for the SC4.*

- Find problems or anomalies in an algorithm. For example observe figure 3, where the curve is not straight but moves in undulating patterns. The frequency of the patterns is not dependent of the frequency of the signal, which points to an irregularity somewhere in the algorithm (or maybe even the host, though the same behavior was observed in three hosts).
- Information about the compressor's knee can be obtained by comparing the gain reduction levels for signals with level close to the threshold and by analyzing data from the parts with rising signal level.
- Other interesting effects (or side-effects) can be studied, like the relationship between the phase of the input signal and the gain reduction applied. This is particularly relevant for lower frequencies, and can be seen on Figure 4, where the oscillation in the gain occurs twice per wave period (at 50 Hz, the period is 20ms).
- The curvature and slope of the gain reduction and gain recover curves determine the sound of the compressor. In figure 8, we can see the gain recovery shape for the SC1 plug-in.

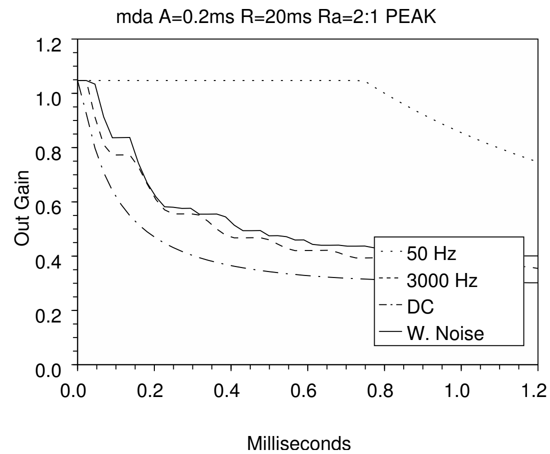

Figure 6: *Gain reduction (PEAK) for the mda plug-in at different frequencies.*

- The comparison of actual characteristics with subjective evaluation might yield information to dictate desirable characteristics of a compressor. For example, the (subjective) higher quality output of compressors like the Waves or R-compressor suggest that Latency and irregularities in the gain curve are undesirable artifacts that should be avoided.
- As can be seen in figure 9, compressors behave very differently even at the same settings. The amount of steady gain reduction for the same threshold and ratio varies dramatically. The attack time is not exact in any case, but merely a time contant which determines behavior. The actual time required to reach a steady state is between 3 to 6 times the specified setting in this case.

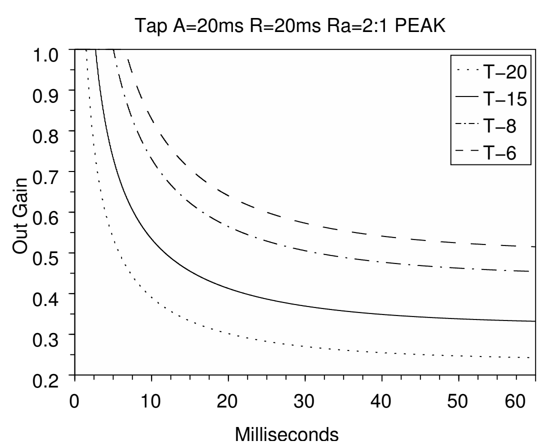

Figure 7: *Gain reduction (PEAK) for the TAP dynamics plug-in at different amplitudes in relation to a threshold (for 1500Hz).*

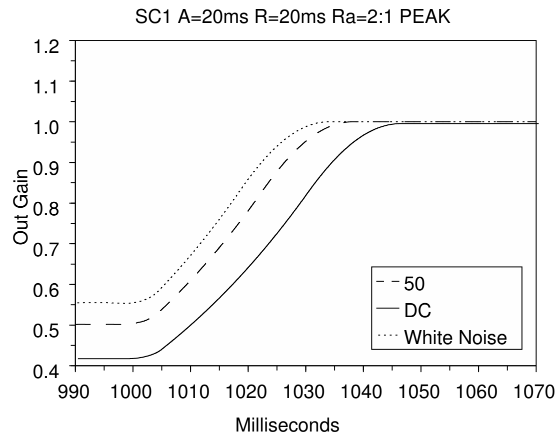

Figure 8: *Recovery curve (PEAK) for the SC1 plug-in at different frequencies.*

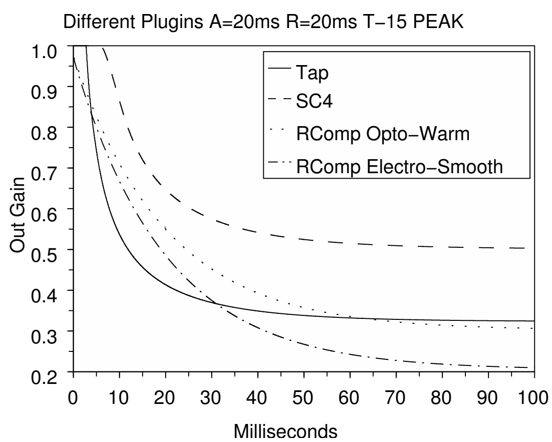

Figure 9: *Comparison of gain reduction for different plugins at the same settings (for 1500Hz).*

3. FUTURE DIRECTIONS

The method will be applied systematically to as much plug-in compressors as possible. In this process, data about the compressors will be gathered and the method will be further refined, leaving out redundant or unnecessary analysis. The method's goal is to provide a deep understanding of compression and a framework for efficient analysis of dynamic processing, to be able to design, emulate and improve dynamic range compressors. With some adjustments, this methodology can also be eventually applied to other dynamic processes like limiting, gate and expansion. The model will also serve to compare a model plug-in with the original by using a different test sample, probably involving musical material, to compare the treatment between the two models.

The perceived quality of the compressor can be compared with the measured results to determine which physical characteristics relate to desirable auditory output. By carefully analyzing a compressor's output, its method of operation can eventually be derived or closely approximated. The analysis should serve as the basis for compressor modeling of both digital and analog processors in the digital domain. It should also serve as a basis for developing novel models that exhibit desirable behavior not present in extant compressors. Part of what defines a compressor is the way it tracks amplitude. Using data provided by this analysis can provide mathematical models for emulation. In addition, the slope and curvature of the gain reduction curves can be measured, to be modeled as well.

Using STFT analysis (or measuring the impulse response) to determine if the compressor applies some sort of EQ on the signal has not yet been implemented in Scilab, but tests using other software have shown flat spectrum on the output, indicating no coloring at all. However the process described here can be also be applied to analog compressors, where this can be significant (see [7]).

4. CONCLUSIONS

Presented here is an efficient method which permits detailed analysis of the effects of plug-in compressors on a carefully prepared test sample. The method at its initial stages has already provided insight into the characteristics of compression.

5. REFERENCES

[1] Waves, "Renaissance compressor," Retrieved June 29th, 2006, [Online] http://www.waves.com.

[2] Cakewalk, "Sonitus fx:Compressor," Retrieved June 29th, 2006, [Online] http://www.cakewalk.com/Products/Sonitus/sonitus.asp.

[3] S. Harris, "SC4 Compressor," Retrieved June 29th, 2006, [Online] http://www.plugin.org.uk/ladspa-swh/docs/ladspa-swh.html.

[4] ——, "SC1 Compressor," Retrieved June 29th, 2006, [Online] http://www.plugin.org.uk/ladspa-swh/docs/ladspa-swh.html.

[5] T. Szilagyi, "Tap Dynamics," Retrieved June 29th, 2006, [Online] http://tap-plugins.sourceforge.net/ladspa/dynamics.html.

[6] Maxim digital audio, "mda dynamics," Retrieved June 29th, 2006, [Online] http://www.mda-vst.com/.

[7] M. J. Kemp, "Analysis and Simulation of Analogue Dynamic Compressors and Limiters in the Digital Domain," in *Proc. 109th AES Convention*, Los Angeles, 2000, preprint 5185.

[8] J. Peissig, J. Remmer ter Hasborg, F. Keiler, and U. Zoelzer, "Digital Emulation of Analog Companding Algorithms for FM Radio Transmission," in *Proc. Int. Conf. on Digital Audio Effects (DAFx-04),* Naples, Italy, 2004, pp. 285–290.

[9] Unique Recording Software, "URS Plugins," Retrieved June 29th, 2006, [Online] http://www.ursplugins.com/.

[10] Python, "Python programming langauge," Retrieved June 29th, 2006, [Online] http://www.python.org/.

[11] R. Boulanger, *The Csound Book*. MIT Press, 2000.

[12] J. Schimmel, "Using nonlinear amplifier simulation in dynamic range controllers," in *Proc. Int. Conf. on Digital Audio Effects (DAFx-03),* London, UK, 2003, pp. 49–52.

[13] INRIA, "Scilab," Retrieved June 29th, 2006, [Online] http://www.scilab.org/.