Queen Mary
University of London

MSc Project Report

A Design of a Digital, Parameter-automated, Dynamic Range Compressor

Name: Dimitrios Giannoulis

Student No.: 090408759

Supervisor: Dr. J.D. Reiss

Date: **26 August 2010**

DISCLAIMER

This report is submitted as part requirement for the degree of MSc at the University of London. It is the product of my own labour except where indicated in the text. The report may be freely copied and distributed provided the source is acknowledged.

Signature:

Date:

DEDICATION

I dedicate my MSc project report work to my family and especially to my parents. I will always appreciate all they did and their words of guidance will always be remembered.

ACKNOWLEDGEMENT

I am heartily thankful to my supervisor, Josh Reiss, for entrusting me with this project and because his guidance and support throughout the year enabled me to develop a deep understanding and tackle any unforeseen obstacles that appeared. I'd also like to thank his research group, and especially Enrique G. Perez, for all the valuable discussions we had and the helpful feedback they offered.

I am also grateful to Michael Massberg since this project would not have been possible without his work on the field. Finally, I would like to offer my thanks and regards to all those who supported me and helped me in any way for the completion of this project.

ABSTRACT

A rather complex and at the same time widely used digital audio effect is Dynamic Range Compression. The versatility of the effect together with the gigantic number of choices regarding its use have made it perhaps the most misused and overused tool in mixing [1]. The parameters of the effect are highly correlated and even though the influences on the signal are not always obvious and distinguishable, the consequences on the mix are profound. All these things make clear that dynamic range compression is a hard to master effect and usually an incorrectly configured compressor will alter the nature of the mix in unpredictable ways introducing a number of various artefacts to the sound, possibly unpleasant.

In this report, we expand on any work previously done on the field of automating the effect of dynamic range compression so as most of its parameters will be configured automatically based on side information extracted by the input signal and the user-adjustable controls and interaction will be kept to a minimum. We try to optimize the method proposed by M. Massberg in his last year report [2] and also investigate alternative automation methods and their impact on the compressor, based on the results of the evaluation tests, he performed last year, comparing the automated compressor settings against preferred human operators' options.

TABLE OF CONTENT

CHAPTER 1: INTRODUCTION

One of the most widely used tools in mixing is the dynamic range compressor. Dynamic Range Compression (DRC) is the process of mapping the dynamic range of an audio signal to a smaller range [3]. That is, reducing the signal level of the higher peaks while leaving the quieter parts untreated so as to narrow the difference between the high and low audio levels of the signal. Whether it is mixing a recording to adjust the playback levels of the various channels or loudness control in television broadcast between programs and channels, an automatic volume control is always asked for and dynamic range compression is exactly that. The DRC process decreases the maximum volume but at the same time leaves quieter sounds and more importantly the noise floor at its original level. Most of the audio signal's characteristics will remain unchanged and only its dynamic range will be reduced. DRC is used extensively in audio recording, production work, noise reduction, broadcasting and live performance applications [4].

Over the years people started overusing and even misusing compression. Sound engineers in their attempt to make sounds louder, bigger and punchier have resorted to more and more compression (Loudness War)[5]. But a superfluously used compressor suppresses the musical dynamics of the sound and this results to lifeless or even boring recordings deprived of their natural sound. Mastering dynamic range compression and refraining from overusing it is not an easy task even for professional engineers [1].

1.1 Compressor Main and Additional Controls

A compressor has a set of controls directly linked to compressor parameters through which one can set up the effect. There are four main compressor parameters. Below, we give the definition for each parameter. In the background section we will present each parameter in detail.

Threshold defines the level above which compression starts. Any signal overshooting the threshold will be reduced in level**.**

Ratio controls the input/output ratio for signals overshooting the threshold level. It determines the amount of compression.

Attack and release Times - also known as time constants - provide a degree of control over how quickly a compressor acts. Instantaneous compressor response is not sought because it introduces intermodulation distortion on the signal. Intermodulation distortion is the distortion of the signal that happens because of excessive gain modulation. To avoid that, we need to slow down the compressors response and this is achieved through the time constants. The attack time defines the time it takes the compressor to decrease the gain to the level determined by the ratio once the signal overshoots the threshold. The release time defines the time it takes to bring the gain back up to the normal level once the signal has fallen below the threshold.

Apart from the main controls a compressor has a set of additional controls most of which are found in most modern compressor designs. These include a Hold parameter, Soft or Hard Knees, Look-Ahead and many more. In this paper we make use of the make-up Gain and the Soft/Hard knee parameters therefore, we will talk about these two:

A *Make-Up Gain* control is usually provided at the compressor output. The compressor reduces the level (gain) of the signal so feeding back a make-up gain to the signal allows for matching the input and output loudness level.

The *Hard/Soft Knee* option controls whether the bend in the response curve has a sharp angle or has a rounded edge. The Knee is the threshold-determined point where the input-output ratio changes from unity to a set ratio. A sharp transition is called a Hard Knee and provides a more noticeable compression. A softer transition where the ratio gradually grows from 1:1 to a set value in a transition region on both sides of the threshold is called a Soft Knee. It makes the compression effect less perceptible. Depending on the signal one can use hard or soft knee, with the later being preferred when we want transparent compression (as in Vocals).

Setting up a compressor's parameters can be proven a laborious task. Changes in one parameter affect the others and the effects of most of the parameters on the signal are not obvious without careful listening. Furthermore, a compressor is a very versatile tool used for a plethora of applications, many of which have opposing natures (e.g. softening transients or emphasizing transients) [1]. If a compressor is not properly set up it will not only provide unsatisfactory results but it will also introduce unpleasant artefacts in the sound, such as alter the natural attack and decay of instruments etc. These are mainly related to the attack and release times also known as time constants. Nevertheless, a badly configured threshold and ratio will also result in unsatisfactory compression, whether that is overcompression or less than what was needed, while a non-accurate make-up gain will result in a level mismatch between the compressor input and output.

For a very short attack time an instrument's natural attack will be suppressed reducing the instrument's punch and clarity. There are cases where that is sought (e.g. drum hits), but in others we want the natural attack of an instrument to be left unchanged. Another consequence of fast attack times is low frequency distortion. The reason being that low frequencies have long enough period for the compressor to act within a single cycle rather than the overall dynamic envelope [1]. On the other hand, a longer attack time than what is required is rarely beneficial. If the attack time is longer than the instrument’s natural attack the dynamic envelope succeeding the natural attack will be altered by the still-increasing gain reduction, resulting in timbre alteration.

Similarly, a very short release time can also distort low frequencies for the same reason short attack can and it can also produce audible clicks. Short release can also affect the silent sections of a signal and cause “pumping”, a compressor artefact of which we will speak more about, later in the paper. On the other hand, longer release times than what is needed can affect the signal in various ways. It may reshape the decay part of notes or cause dropouts after very short transients because the compressor takes too long to bring back the level.

1.2 The Automation Process

Automating the Compressor parameters and in general, the parameters of any audio effect, can provide evident advantages to the user. It will save amateur users the trouble of properly setting the effect to avoid sound artefacts and in most cases it will give better results. Professional users will probably see no direct advantage out of the use of an automated effect but still there are cases where an automated effect can come at hand. A highly diverse signal, whether that is a single instrument with a varied playing technique and sound or a mix containing several instruments, will not be optimally compressed using a static set of parameter settings. An automated compressor with constantly adapting parameters to the signal’s characteristics will be able to give better results due to its adaptability. The requirement of such effects in the market is already apparent and that is why compressors with partly automated parameters (like auto release) have already found their way to production both as analogue and digital designs [1]. Furthermore, research on the field of automating the compression effect goes back many years [6] and is still active [2].

1.3 Improving Existing Automation Methods

This project is expanding on the work on Automated Dynamic Range compression, done last year by Michael Massberg as part of his MSc project “Investigation in Dynamic Range Compression” [2]. Massberg in his work managed to successfully automate the attack and release times the make-up gain and the knee width of the compressor using an intelligent way that depended on the signal’s level. The results, although not the most favourable, were acceptable and the automated compressor he developed could perform satisfactory compression in most music signals from vocals to percussive sounds. Limitations of the method involved that it did not try to extract and make use of any properties and statistics of the input signal other than the crest factor of which we will speak in chapter 3. The aim of this project is to optimize the automation methods used in “Investigation on Dynamic Range Compression”, investigate alternatives and improve the overall automation process.

We investigate on alternative methods to reach proper automation of the attack and release times like the use of onset detection functions like the Spectral Flux. We try to improve the make-up gain by introducing a proper loudness model that compares the loudness of the signal before and after compression. We try and extract information on the transient content of the signal and use them to adjust the knee type and the knee width, in contrast to the more static implementation of the knee width Massberg created, where the width was analogous to the amount of compression.

CHAPTER 2: BACKGROUND

Compression is one of the most common processes in audio and music mixing, yet compressors are one of the most misused audio effect processors. This is mainly due to the fact that compression is a highly non-linear effect and has to be used with care since it may alter a signal in unpredictable and not desired ways [1], [4]. Because of that, and because of the numerous ways one can use the effect, there are countless compressor designs. No two compressors sound alike, certainly not two analogue compressors [1]. Each one is inaccurate in its own unique way. Some differ in the compressor topology, others introduce additional stages and some simply differ from the precise digital design since these deviations add character to the compressor. Nevertheless, there is a set of standard stages that are found in all the designs and even though they might differ slightly from one another they all follow the same principals.

2.1 Principle of Operation

In order to understand compression we need to look at the main stages within a compressor and its internal building blocks.

The signal entering the compressor is split in two copies. One is sent to a variable-gain amplifier and the other to a side-chain where a circuit controlled by the level of the input signal applies the required gain reduction to the gain stage. There are two possible topologies: a **feedback** type and a **feed-forward** type topology (see figure 1).

In the feedback topology the input to the side-chain is taken after the gain stage. This was traditionally used in early compressors and had the benefit that the side-chain could rectify possible inaccuracies of the gain stage. However the design has a few limitations like the inability to allow a look-ahead function or to work as a perfect limiter due to the infinite negative amplification needed to achieve that [2].

The feed-forward topology has the side-chain input before the gain stage. This means that the side-chain circuit responsible for calculating the gain reduction to be applied to the signal gain, the gain computer, will be fed with the input signal. Therefore it will have to be accurate over the whole signal's dynamic range as opposed to a feedback type compressor where it will have to be accurate over a reduced dynamic range since the side-chain is fed with the compressor's output. This is not as much a problem as a special consideration when designing an analogue feed-forward compressor. It bears no implications to a digital design. Most modern compressors are based on the feed-forward design, thus we will also choose to implement this design.

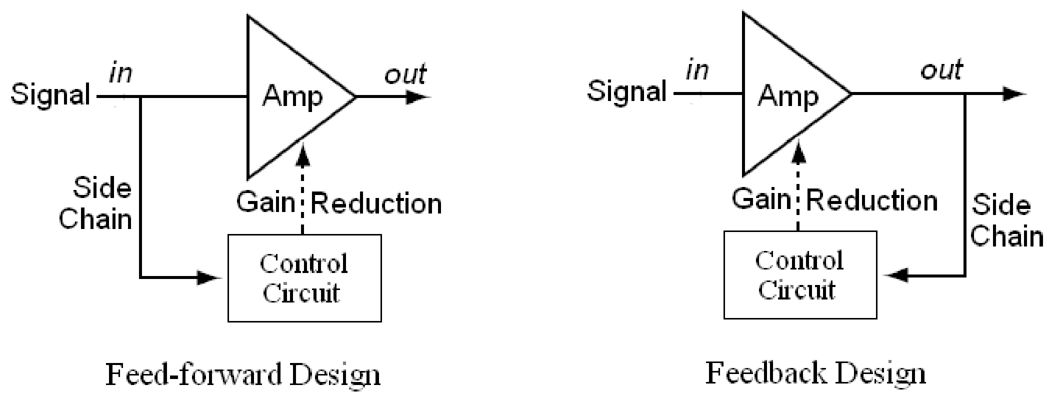


Figure 1. Feedback and Feed-forward type compressor designs.

2.2 The Gain stage

As we have said already, the gain stage is responsible for attenuating the input signal by a varied amount over time of decibels (dB) determined by the side-chain. The most widely used type of analogue compressor gain stage is a solid-state Voltage-controlled amplifier (VCA), because they provide the most accurate and controllable gain manipulation. In a VCA an external control voltage ($\boldsymbol{cv}$) coming from the side-chain determines the varied gain reduction the VCA will apply to the input signal. Voltage control amplifiers can be seen in the feed-forward and feedback designs in figure 1. Other compressor designs might include FETs, Vari-mu tubes or optical compressors instead of VCAs.

In a solely digital design, one can model an ideal VCA using a set of mathematical operations involving multiplying the input signal by the exponent of the control voltage ($\boldsymbol{cv}$) coming from the sidechain [2].

$$\mathrm{out} = \mathrm{in} \cdot \exp(\mathrm{cv}) \tag{1}$$

And taking the logarithm of both sides:

$$\log|\mathrm{out}| = \log|\mathrm{in}| + \mathrm{cv} \tag{2}$$

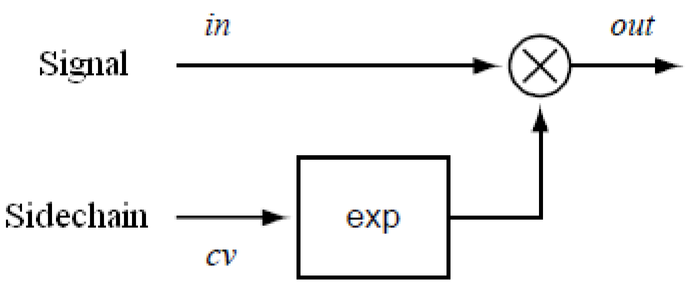


Figure 2. Block diagram of a digital implementation of a VCA

2.3 The Level Detection stage

The first stage the signal encounters as it enters the side-chain is the level detection stage. In that stage, the signal's bipolar amplitude is converted into a unipolar representation of level. In this case the level of the signal is determined by its peak value (instantaneous signal level) and this is known as peak-sensing. An alternative would be to use RMS-sensing and determine the signal level by its RMS value. RMS-sensing is closely related to the loudness of the incoming signal rather than its peak value. Compressors can use either method or support a toggle option between the two.

2.4 The Gain computer stage

The gain computer is the compressor stage that generates the control voltage. $\boldsymbol{cv}$ determines the gain reduction to be applied to the signal. This stage involves the compressor's **Threshold** and **Ratio** parameters. These define the static input-to-output characteristic of compression. Once the signal level exceeds the threshold value, it is attenuated according to the ratio. In figures 3 and 4, we present typical compression gain curves that illustrate the threshold and ratio features.

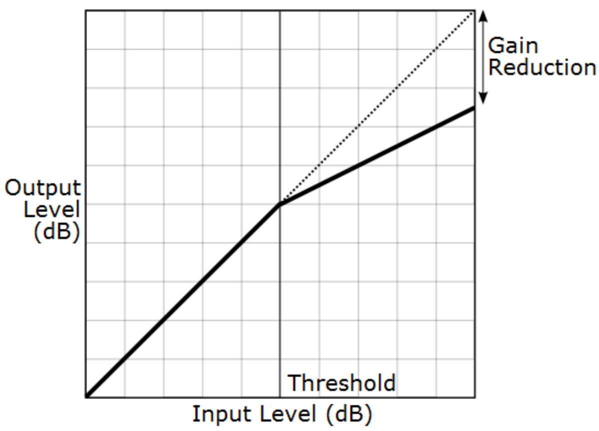


Figure 3. A typical Compression Gain Curve

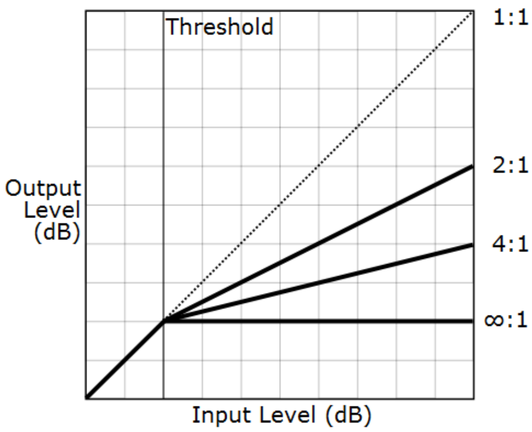


Figure 4. Compression Gain Curves for various ratios

From figures 3 and 4 and some simple linear algebra we end up with the following relationship that takes into account the change in slope of the input/output compression curve:

$$\log|\text{out}| = \begin{cases} \log|\text{thr}| + \dfrac{\log|\text{in}| - \log|\text{thr}|}{\text{ratio}} & \text{for } \log|\text{in}| > log|\text{thr}| \\ \log|\text{in}| & \text{otherwise} \end{cases}$$

Where **thr** stands for Threshold.

Substituting $\mathbf{log|in|}$ **with** $\mathbf{log|out| - cv}$ from Eq.2 in the above equation like we have done, we get a formula for the control voltage for a feed-forward type compressor:

$$\mathbf{cv} = \begin{cases} \mathbf{-\log|in| + \log|thr| + \dfrac{\log|in| - \log|thr|}{ratio}} & \textbf{for } \mathbf{\log|in|} > \boldsymbol{log}\mathbf{|thr|} \\ \mathbf{0} & \textbf{otherwise} \end{cases}$$

$$\mathbf{cv} = \begin{cases} \mathbf{\log|in| - \log|thr|} \left(\mathbf{-1 + \dfrac{1}{ratio}}\right) & \textbf{for } \mathbf{\log|in|} > \boldsymbol{log}\mathbf{|thr|} \\ \mathbf{0} & \textbf{otherwise} \end{cases}$$

$$\text{cv} = \text{slope} \times \max(\log|\text{in}| - \log|\text{thr}|, 0) \quad \text{with} \quad \text{slope} = \frac{1}{\text{ratio}} - 1$$

2.4.1 Digital Implementation of the Gain Computer

A half-wave rectifier applied on the difference between the log-encoded threshold and input signal level and multiplying the rectifier's output by the slope variable gives us the control voltage for the digital implementation.

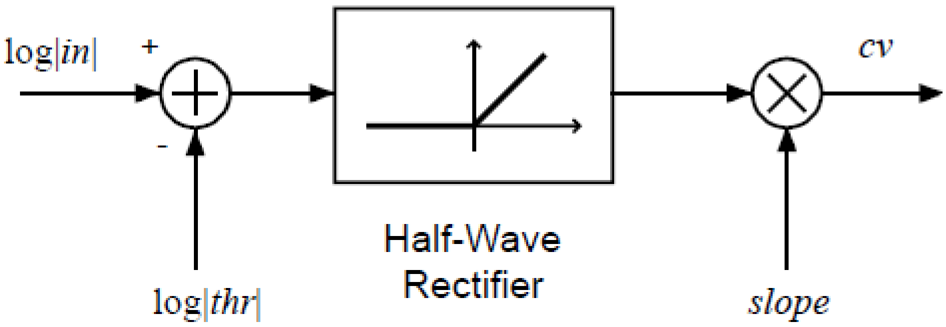


Figure 4. Gain Computer Block diagram

2.4.2 Soft Knee Implementation

So far we have only talked about hard knee implementations. A soft knee implementation will provide a smooth transition band at the threshold point. The width of this band known as knee width will equally extend to both sides of the threshold, the lower with slope of 1 and the upper with slope equal to 1/ratio. Figure 5 presents a compression gain curve with a soft knee.

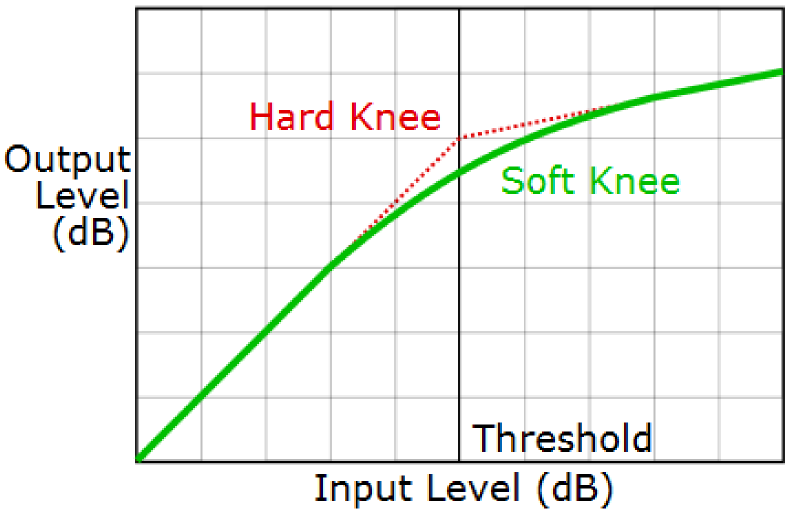


Figure 5. Compression Gain Curve with Hard/Soft Knee

To implement this we need to replace the half-wave rectifier from 2.4.1 with a soft rectifier function $\mathbf{f}(\boldsymbol{x})$, as presented in figure 6, and ask for continuity at the knee width borders [2]. This is illustrated in figure 6 below.

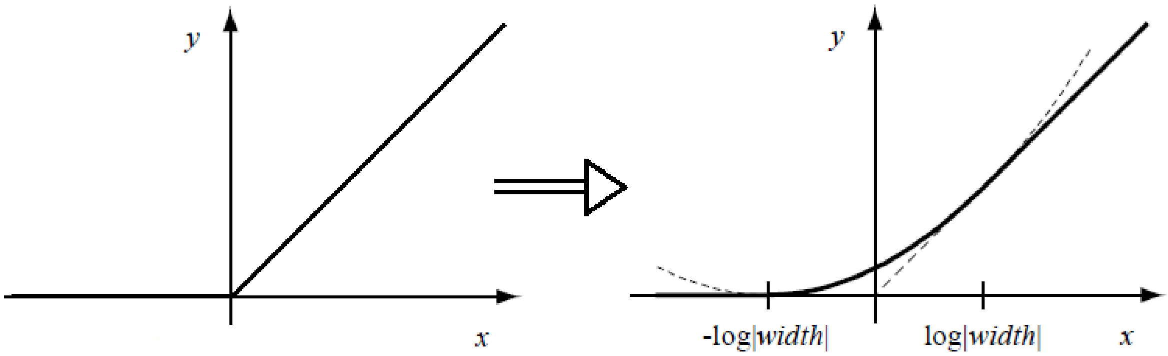


Figure 6. Substituting the Half-wave rectifier with a smooth (soft knee) rectifier

The control voltage now will be defined as:

$$\mathrm{cv} = \mathrm{slope} \times \mathrm{f}(\log|\mathrm{in}| - \log|\mathrm{thr}|) \quad \text{with} \quad \mathrm{slope} = \frac{1}{\mathrm{ratio}} - 1$$

$$\mathrm{f}(\mathrm{x}) = \begin{cases} 0 & \text{for } \mathrm{x} < -\log|\mathrm{width}|/2 \\ \dfrac{1}{2\log|\mathrm{width}|}\left(\mathrm{x} + \dfrac{\log|\mathrm{width}|}{2}\right)^2 & \text{for } -\log|\mathrm{width}|/2 \le \mathrm{x} < \log|\mathrm{width}|/2 \\ \mathrm{x} & \text{for } \mathrm{x} \ge \log|\mathrm{width}|/2 \end{cases}$$

The soft knee function, we have called f(x), is presented analytically in “Investigation on Dynamic Range Compression” [2] so there is no need to analyze it in here as well.

2.5 Attack and Release Times

We have covered the process of how the control voltage is calculated and how this leads to a gain reduction on the input signal. But we have not referred to the fact that the change of gain is applied smoothly and over some time rather than instantaneously. The gradual change of gain is due to the attack and release times that are usually introduced in the compressor's circuit through a smoothing detector filter. In figure 7, we present the compressor's behaviour during the attack and the release phases.

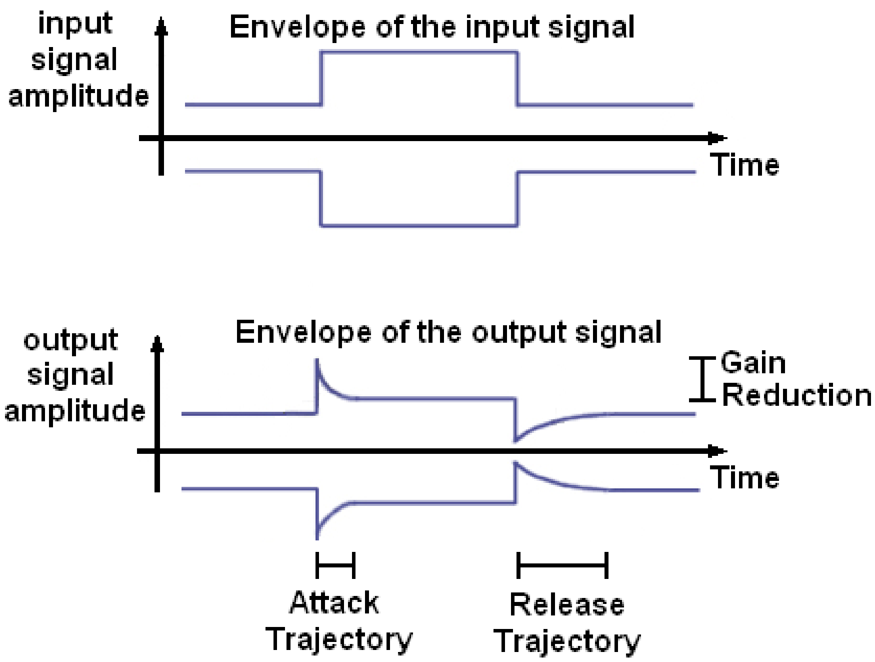


Figure 7. The attack and the release phases in a compressor

There is quite a lot bibliography about different analogue envelope detector designs [7]. In "Investigation on Dynamic Range Compression", Massberg performed a thorough study on the different designs and arrived at the conclusion that the most preferred detector is the decoupled peak detector (figure 8). Because of the topology of the decoupled peak detector the attack envelope is impressed upon the release envelope. In other words, the attack time is added to the release time. This guarantees that the release time can never be shorter than the attack time, and that can be seen as quite a useful property if both times are to be heavily automated and we do not want to end up with a shorter attack time than release time. [2]

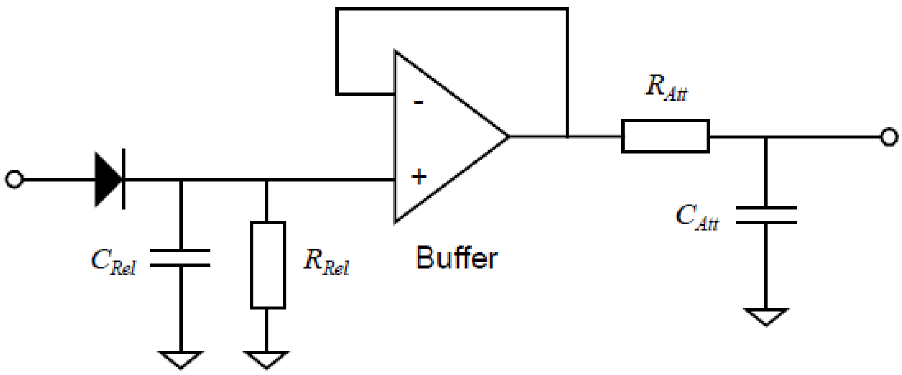


Figure 8. Decoupled peak detector circuit

Picture taken from Michael Massberg's "Investigation on Dynamic Range Compression" [1]

$$y_{Rel}[n] = \max(x[n], y_{Rel}[n-1] - \alpha_{Rel} y_{Rel}[n-1])$$
$$y[n] = y[n-1] + \alpha_{Att}(y_{Rel}[n] - y[n-1])$$

Where α_{Att} and α_{Rel} are coefficients calculated from τ_{Att} and τ_{Rel} respectively.

$$\tau_{Att,Rel} = R_{Att,Rel} C_{Att,Rel}$$

Similarly there are various preferred placements of the detector circuit inside the compressor's circuitry. In some papers the suggested position is after the gain computer in the linear domain [8], in others within the linear domain in front of the gain computer [7] etc. In the investigation on compressors that Massberg conducted the most preferred position for the detector to be placed was found to be in the log domain after the gain computer, since in this position a smooth envelope is generated with no attack lag and the capability of easily using a variable soft knee. [2]

2.6 The Make-up Gain stage

The purpose of the make-up gain as this was explained in the introduction of the paper is to add a constant gain back to the signal in order to match output and input levels since the output will have a decreased gain level because of the compression. Without the make-up gain the output signal (compressed) will always sound quieter (less loud) than the input signal (uncompressed). A less loud sound will most likely be perceived by the human ear as inferior to a louder sound. That is because the human auditory system has the tendency to conceive louder sounds as superior to softer ones even if they are identical in every other aspect [9]. Because of that the make-up gain control has become a standard in the compressor's design and it is quite hard to find a modern compressor without this function. In a digital compressor we can easily implement a make-up gain by multiplying the compressor's output by a constant factor corresponding to the desired make-up gain value.

2.7 The Look-ahead function

An optional function found in many compressors is the look-ahead function. Compression can prove to be quite tricky with signals containing sharp level changes like in the case of transients. In order to catch those transients a compressor has to have a very fast response. But a very fast response can produce audible artefacts like pumping and might not produce musical results. It would be ideal if the side-chain could see the signal input slightly in advance of it reaching the VCA so it would give the compressor more time to react to transients. This is achieved with the use of the look-ahead function.

The look-ahead is implemented by introducing a delay unit (of a few ms) after the signal copy that is sent to the side-chain and before the variable-controlled amplifier. This way the input will be seen immediately by the side-chain circuit but it will be processed by the VCA shortly after. This will introduce an output delay but in most cases it will not be perceived, even though it might lead to phase mismatch issues. Auto delay compensation can make up for this delay of the output.

In the case of an automated compressor like ours, a look-ahead is not necessary since the effect is constantly and automatically adaptable based on the signal's characteristics and features. Nevertheless, it might improve the overall performance.

CHAPTER 3: DESIGN AND IMPLEMENTATION

After carefully examining the various design possibilities and options we decide on a final compressor configuration that we will use as the base of our model. Later we will introduce various automated options on the configuration to complete the development of the automated compressor version.

3.1 Structure of the Compressor Model

The configuration process of the compressor model involves a series of choices over specific designs. The very first one involves a choice over the compressor type and therefore, the compressor’s general topology. We choose to work with a feed-forward compressor as our design is digital and we do not have to worry for accuracy problems over a high dynamic range as these could appear in an analogue design. Feed-forward compressors perform better overall, because they are more stable and predictable than the feedback type ones [2].

We want the compressor to have a smooth performance as much as possible in order to be able to perform well on a variety of signals from vocals to drums. A smooth performance will minimize the introduction of unwanted artefacts in the sound, as well as possible alterations on the natural characteristic of each instrument. As discussed above, the preferred envelope detector design to provide this kind of smooth performance is the decoupled peak detector design. The detector is placed in the log domain and after the gain computer, since as we mentioned in chapter 2 this place is the best in providing a smooth envelope without any attack lag.

Furthermore, we will include a soft knee with a variable width and a make-up gain stage. The result will be a full featured compressor for general use that can perform well or just acceptably under the worst conditions. It is worth mentioning that most compressors are built in such ways that usually perform well over specific audio signals and instruments and not that well in all other cases [1].

Figure 9 depicts the compressor configuration and figure 10 depicts the block diagram of the compressor model.

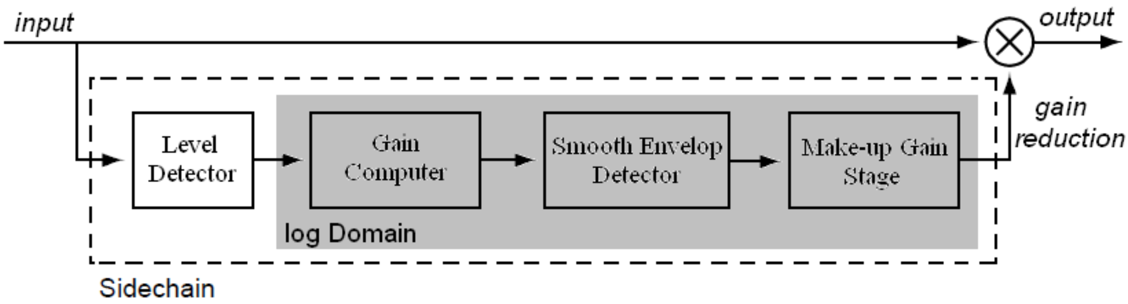


Figure 9. Compressor Model

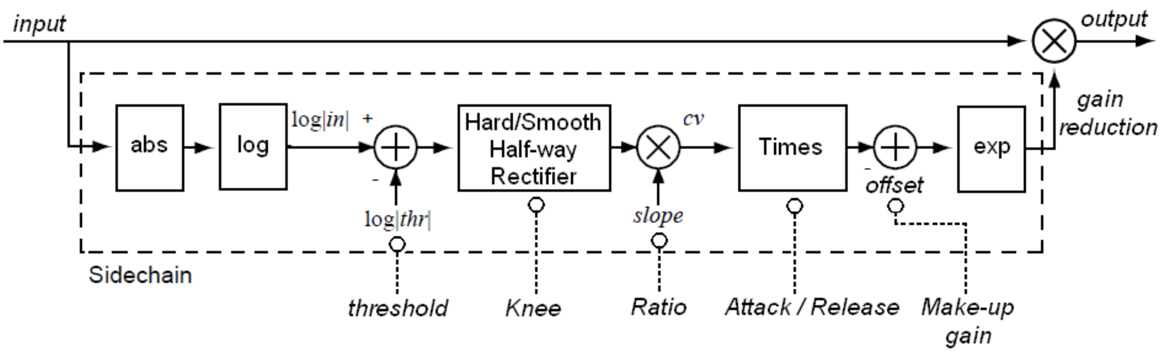


Figure 10. Block Diagram of the Compressor Configuration

3.2 Parameter Automation

The compressor model, presented in figure 9, takes 5 input parameters as compression settings. In this paper we will investigate and develop methods to automate each parameter independently so there will be no need to set it up when using the effect. As these parameters are used in different stages of the compressor design, their automation methods will be independent of each other, even though they might be based on the same signal statistics at some cases.

3.3 Auto Attack and Release Times

Automating the attack and release times in a compressor is the first step towards automating the effect. Since most of the signals are not static in nature and have an ever changing dynamic content, at times full of transients and at times mainly steady-state, automatically adjustable time constants will perform better than static set up ones. There have started appearing compressors with auto attack and auto release functions in the market, and their increasing popularity verifies the benefits of such functions [1]. In most designs, the automatic computation of the time constants is done by observing the difference between the peak and RMS levels of signal fed in the side-chain. The method provides good results and at least in the case of the release time works extremely well. Before going into analysing the various automating methods we used, we should present the difficulties related to the set up of the attack and release times.

3.3.1 Artefacts associated with the Time Constants

In section 1.1 we mentioned the need for properly set up time constants and presented the difficulties associated with the set up process. Very short attack and release times should be avoided because they introduce a number of unpleasant artefacts like pumping and breathing, two artefacts related to the use of dynamic range compressors.

Pumping is caused when the compression effect (the gain reduction) is obvious to the listener. This is usually because of quick noticeable level variations such as fast level drops after short transients that surpass the threshold level. The phenomenon is associated with loud level variations and is more perceived in heavy compression or limiting [1] [4].

Breathing is less extreme than pumping and is caused by varying the noise level (hiss) of a signal with high noise content. This may cause an audible airy sound similar to breathing. It is usually noticeable in the quiet portions of the signal. As we mentioned in section 1.1 of this paper, other effects associated with short attack and release times can be low frequency distortion, depicted in figure 11, and altering the attack of an instrument, which usually leads to less punch and clarity.

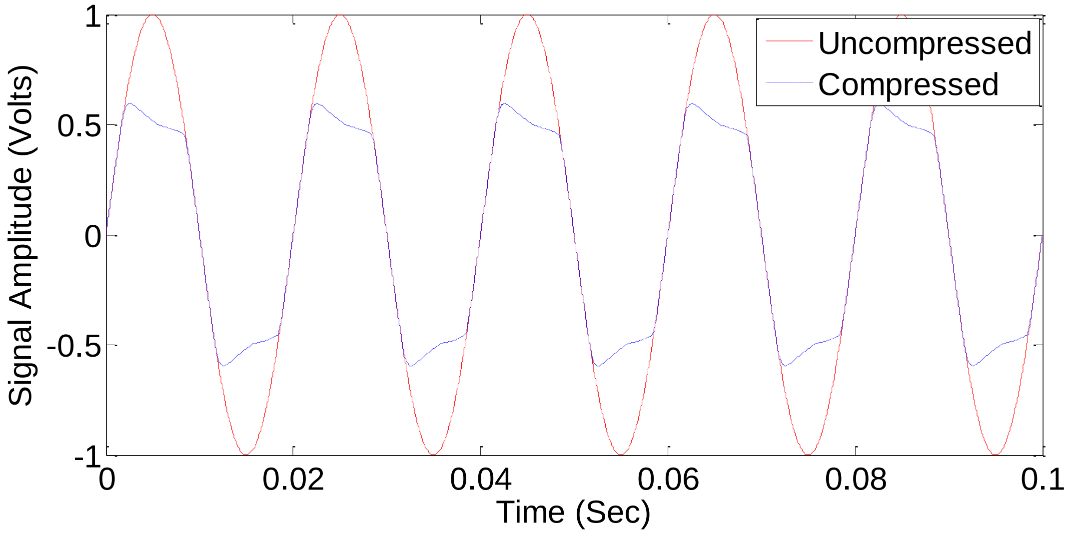


Figure 11. Low Frequency Distortion caused by a fast compressor

On the other hand, very long attack and release times are not welcome either. The longer the attack the less responsive the compressor is to the signal. A slow response can lead to timbre alteration since the long gain reduction effect caused by a long attack time will affect a long part of the signal envelope, as depicted in figure 12. Likewise, a long release time will probably cause perceived dropouts after short transient sounds because of the slow gain restoration, or will alter the decay part of notes and modify the sound of instruments (see figure 13).

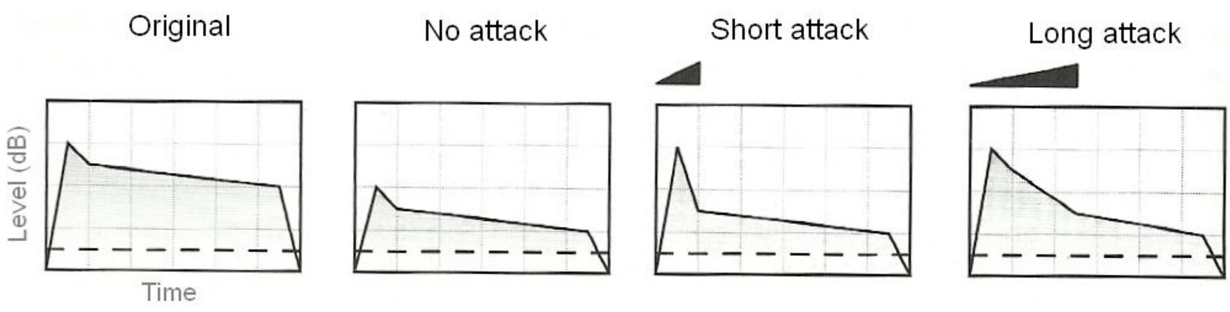


Figure 12. Attack times on a piano key hit - the envelope reshaping phenomenon. Taken from [2].

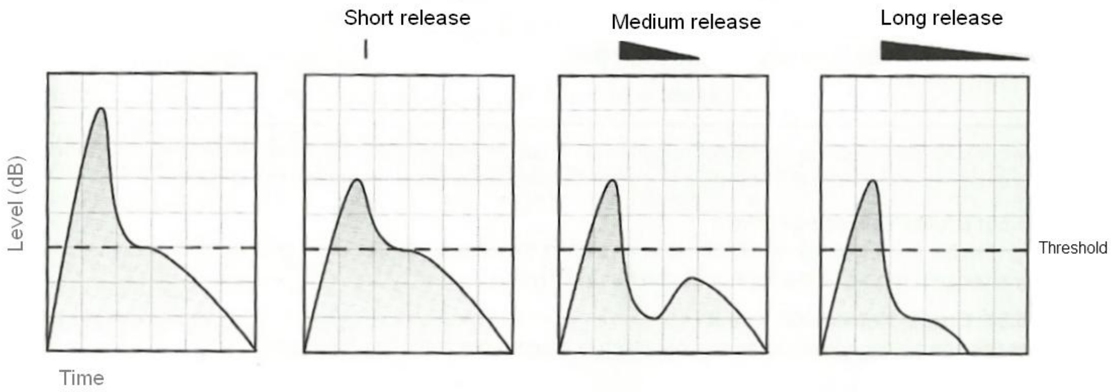


Figure 13. Release times on a snare drum hit – the phenomenon of altering the decay. Taken from [2].

All these artefacts are usually undesirable and engineers try to avoid them as much as possible when using the effect. Nevertheless, dynamic range compression can be used for more creative tasks like other similar nonlinear effect such as noise gates, expanders etc. One can turn the unwanted side effects of the compressor usage to their benefit. For instance having very short attack times, thus not keeping the whole of the natural attack, is a good way to soften drum hits and make them less dominant in a mix. Even the audible clicks caused by extremely short attack times can be used constructively to add definition to the kick drum. Low frequency distortion can add warmth to a mix if it is added in small amounts and pumping can be used in ways to cause the whole mix to alter in volume rhythmically in time with the beat.

The control that attack and release times offer to us, gives us a great degree of freedom over how creative we want to be in the mixing process. Compression can be used in sophisticated ways to bring instruments forward or backward in the mix, apply dynamic movement on a mix with static dynamics, emphasize the decay and so on [1]. All these tasks usually go beyond compressor's main use, which is to reduce the dynamic range by making loud sounds quieter. Therefore, it is hard to create an automated compressor capable of performing creative tasks. And even if we design the automation mechanisms in such a way as to include a creative task, the engineer is the last to decide whether the mix will benefit from the specific creative task or not. In most cases it is just a matter of subjective taste and that is where an automated compressor obviously fails. It would be impossible to guess the artistic intentions of its user and adjust its process respectively.

3.3.2 The Crest Factor Method

Massberg used the crest factor measurement as a method to automate the attack and release times of the automated compressor model. The crest factor, defined as the ratio of peak to RMS, is a useful short term signal measure to determine the nature of the signal. A steady state signal has a low crest factor value, since the RMS value is close to the peak value. For example, the crest factor of a square wave is 1 because both peak and RMS values match each other and the crest factor of a sine wave is $\sqrt{2}$. On the other hand, a signal with a large transient content has a significantly higher crest factor value. Transients usually have high peak values and very short duration. Therefore, they are characterised by low RMS values because they contain small amounts of energy. The high peak value in relation to a small RMS value gives high crest factor values. For that reason, the crest factor can be used as a method to locate transient parts, like note onsets, in the signal.

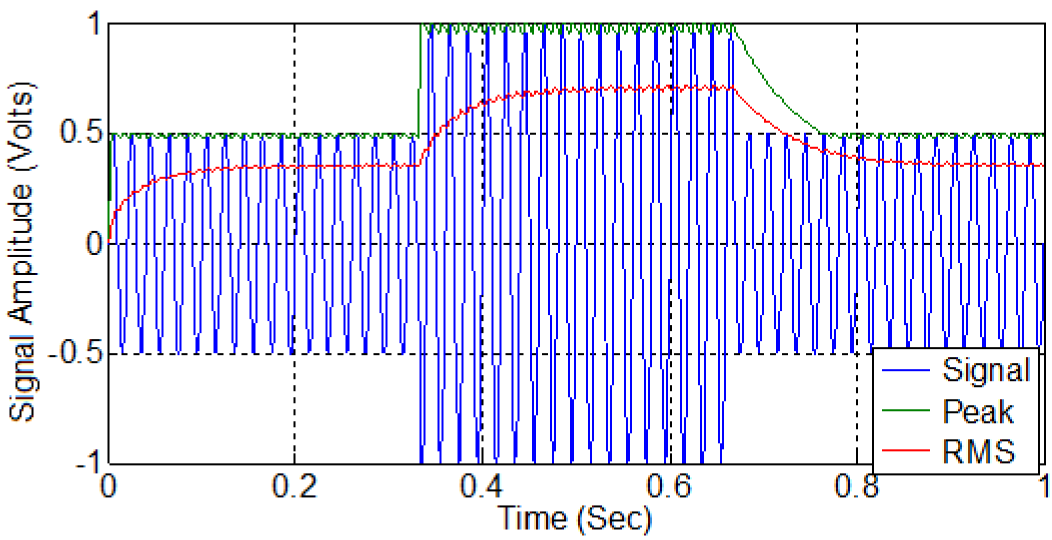


Figure 14. Peak and RMS measurement on sine wave with varied amplitude

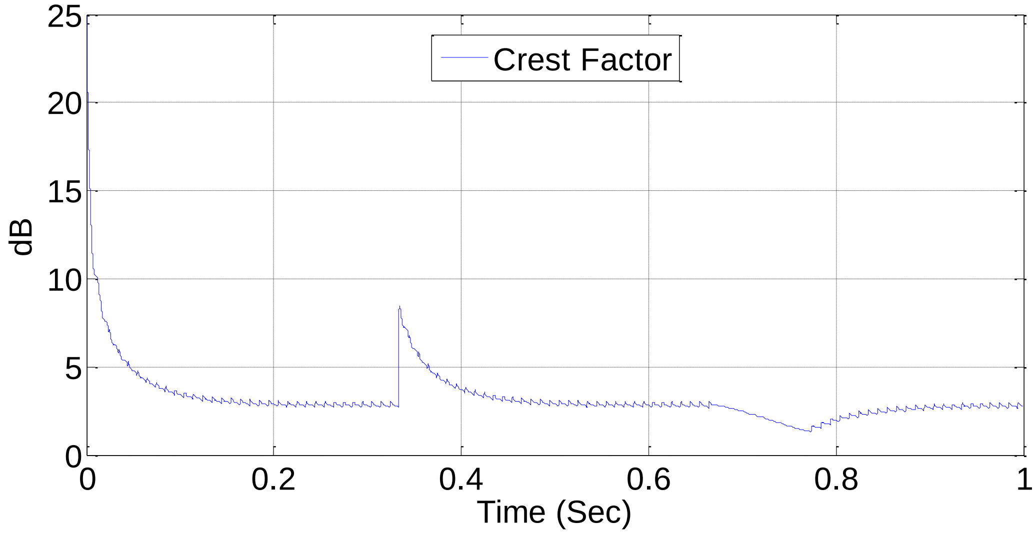


Figure 15. Crest Factor measurement on the sine wave from figure 12.

A signal with a high transient content like the attack part of a note needs short time constants to make the compressor able to catch those transients. On the other hand, a steady-state signal needs lower time constants, thus less noticeable compression, to avoid reshaping its envelope. Hence for a high crest factor, small time constants are required and vice versa. This implies that the crest factor should be analogous to the multiplicative inverse of the attack and release times.

3.3.3 Implementation of the Crest Factor Method

In order to measure the signal’s crest factor value per sample we need to measure the peak value and the RMS value of each sample and divide the two measurements. This can be achieved using a peak detector and an RMS detector as these were presented in “Investigation on Dynamic Range Compression” [2]. The peak detector will have instantaneous attack time and a smooth release trajectory defined by a release time constant. The RMS detector will have an averaging time constant, chosen to be identical to the peak detector’s release time constant. In such a way, we guarantee that both detectors will have the same release envelope and therefore, the peak value will never be below the RMS value [2]. The time constant($\boldsymbol{\tau}$) for the two detectors has to be chosen carefully. We should avoid a very short time because it will result in the RMS measurement following the peaks too closely. But, we should also avoid a very large time because the detectors will fail to react in quick changes of the signal.

After calculating the crest factor value, we need to relate that value to a corresponding attack and release time. The crest factor of a sine wave is $\sqrt{\mathbf{2}}$ ($\approx \mathbf{3dB}$) and can reach high enough values depending on how many transients a signal contains (how larger is the peak value relative to the RMS measurement). If we choose a maximum time constant, one for the attack and one for the release time, multiply it by $\sqrt{\mathbf{2}}$ and divide it by the crest factor measurement we can calculate a signal dependent time constant measurement. That signal dependent measurement will give the maximum attack and release time if the crest factor is that of a sine wave (so $\sqrt{\mathbf{2}}$) but it will give a lot lower values if the signal has more transients.

After testing, Massberg found that regardless of the crest factor measurement's averaging time constant, there was no way the compressor could reach short enough time constants after transients. This resulted in the compressor 'pumping'. An effective solution for this problem was to use the square of the crest factor in the denominator of the equation for calculating the attack and release times, and also multiplies the nominator by two. This made the effect of the crest factor on the time constants more extreme. Furthermore, we should note here that to compensate for the influence of the attack trajectory on the release trajectory in the decoupled peak detector, we have to subtract the attack time from the release [2]. Below we present the final form of the attack and release times equations, as these were used by Massberg in his design.

$$\tau_{Att}[n] = \frac{2\tau_{Att,Maximum}}{{y_{crest}}^2[n]} \qquad \tau_{Rel}[n] = \frac{2\tau_{Rel,Maximum}}{{y_{crest}}^2[n]} - \tau_{Att}[n]$$

Where $\boldsymbol{y_{crest}}$ is the crest factor values per sample $\boldsymbol{n}$, and $\boldsymbol{\tau_{Att,Maximum}}$ and $\boldsymbol{\tau_{Rel,Maximum}}$ are the maximum values for the attack time and the release time respectively and where fine-tuned by Massberg at 80 ms and 1 sec correspondingly after conducting several tests with sine waves [2].

The Crest factor method is an easy, fast and computationally inexpensive way to adjust the time constants relatively to the signal content. The method is very effective and with satisfactory results. Nevertheless, there are other more sophisticated approaches that use time-frequency representation to analyze a signal and detect its transient content [10].

3.3.4 The Spectral Flux Method

Spectral Flux (SF) is a measure of how quickly the power spectrum of a signal is changing. Therefore, Spectral Flux offers detection based on amplitude or energy information of the signal. Other methods exist that are based on phase information or a combination of phase and energy information [10], [11].

The spectral flux method makes use of a time-frequency representation of the signal based on the short time Fourier transform (STFT). The STFT of an input signal x(n) is defined as:

$$\mathbf{X}(\mathbf{n},\mathbf{k}) = \sum_{\mathbf{m}=-\frac{\mathbf{N}}{\mathbf{2}}}^{\frac{\mathbf{N}}{\mathbf{2}}-\mathbf{1}} \mathbf{x}(\mathbf{nh}+\mathbf{m})\boldsymbol{\omega}(\mathbf{m})\mathbf{e}^{-\frac{\mathbf{2j\pi mk}}{\mathbf{N}}}$$

$\boldsymbol{X(n,k)}$ represents the $\boldsymbol{k^{th}}$ frequency bin of the $\boldsymbol{n^{th}}$ frame. $\boldsymbol{\omega(m)}$ is an N-point hamming window and $\boldsymbol{h}$ is the hop size between adjacent windows.

Spectral flux is calculated by comparing the change in magnitude for one frame, thus summed across all frequency bins, against the change from the previous frame. Furthermore, the method is restricted to count only those frequency bins where the energy is increasing (onsets). This is achieved by rectifying the magnitude differences between adjacent frames. The spectral flux function is defined as:

$$\mathbf{SF}(\mathbf{n}) = \sum_{\mathbf{k}=-\frac{\mathbf{N}}{\mathbf{2}}}^{\frac{\mathbf{N}}{\mathbf{2}}-\mathbf{1}} \mathbf{H}(|\mathbf{X}(\mathbf{n},\mathbf{k})| - |\mathbf{X}(\mathbf{n}-\mathbf{1},\mathbf{k})|)$$

Where $H(x) = \frac{x+|x|}{2}$ is the half-wave rectifier function.

L1-norm is favoured over L2-norm for use in the SF based on empirical tests [11]. A further option is the normalized version of the spectral flux where the sum of the magnitudes over all bins in a frame is factored out. This limits the output value of the function between 0 and 1. The normalized spectral flux is defined as:

$$\mathbf{SF_{norm}(n)} = \frac{\sum_{\mathbf{k}=-\frac{\mathbf{N}}{\mathbf{2}}}^{\frac{\mathbf{N}}{\mathbf{2}}-\mathbf{1}} \mathbf{H}(|\mathbf{X}(\mathbf{n},\mathbf{k})| - |\mathbf{X}(\mathbf{n}-\mathbf{1},\mathbf{k})|)}{\sum_{\mathbf{k}=-\frac{\mathbf{N}}{\mathbf{2}}}^{\frac{\mathbf{N}}{\mathbf{2}}-\mathbf{1}} |\mathbf{X}(\mathbf{n},\mathbf{k})|}$$

Spectral flux is typically used for onset detection purposes [11], [12]. Nevertheless, it can easily be used for transient detection purposes since it essentially computes the degree of dynamic behaviour of harmonic partials. That is, if partials stay at the same levels or decays naturally in consecutive frames then the normalized spectral flux will be 0. On the other hand, if harmonic partials fluctuate, as is usually the case in transient parts of the signal like onset of notes the output of the spectral flux function will increase toward 1. Spectral Flux as a transient detection method has been successfully used in other papers [13], [14].

3.3.5 Implementation of the Spectral Flux Method

The spectral flux method will work on the same principle as the crest factor method. The more transient a signal part is, the higher its spectral flux value will be and the shorter the time constants that are needed to achieve proper compression. Therefore the equations that relate the spectral flux measurement to the time constants should be similar to those used for the crest factor. Additionally, this has the advantage that the crest factor method has been already tested with success, so if the added sensitivity of the spectral flux method to transient components is added on the existing model, we expect even better results.

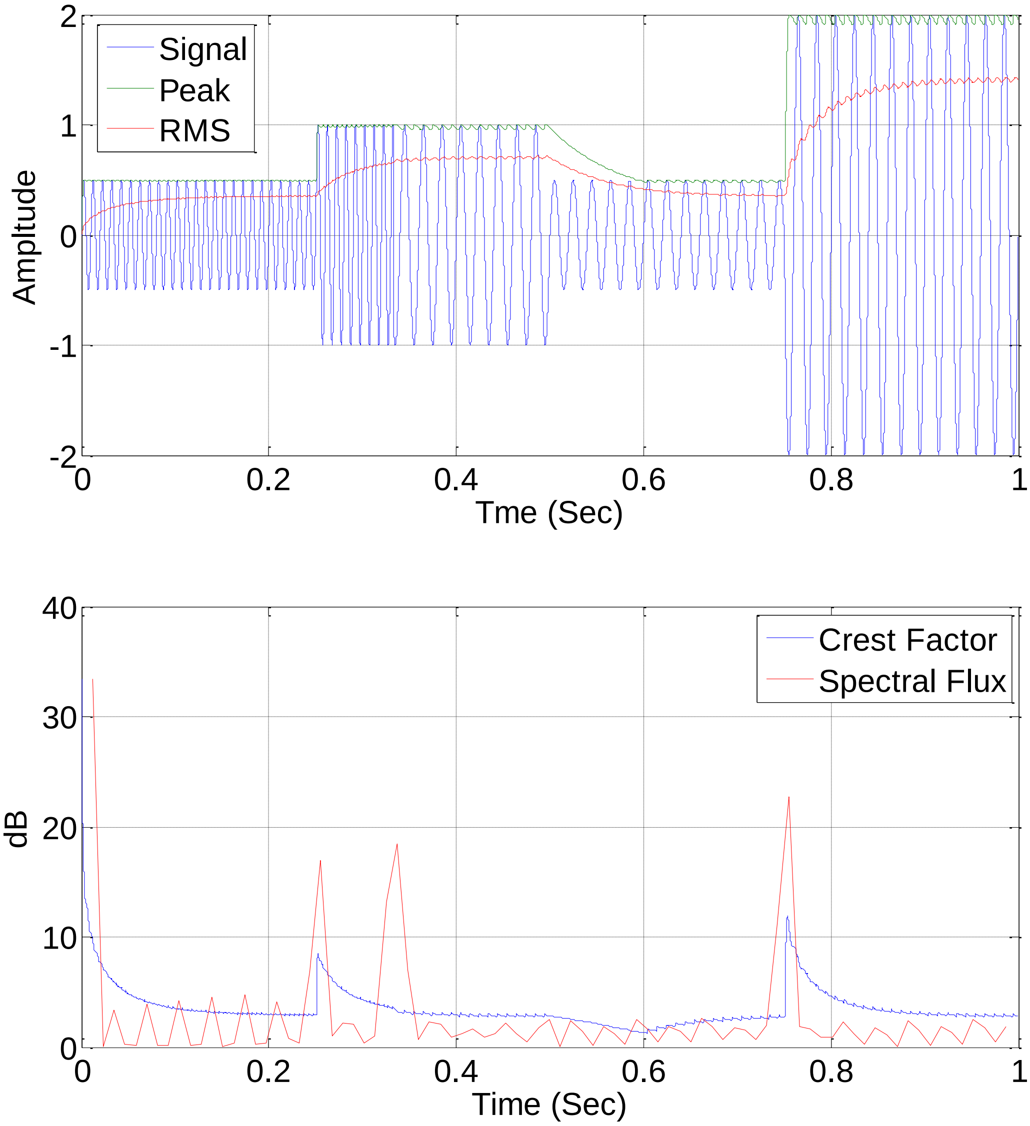


Figure 16. Crest Factor VS Spectral Flux measurement on a sine wave with varied amplitude and frequency

After running many optimization tests for the spectral flux method, we ended up using a window N = 1024 points for the Fourier transform, with a hop size between adjacent windows h = 512, thus 50% overlap between windows. These numbers produce the best figures among other settings, with narrow enough peaks for the spectral flux function, exactly at the same time instances as the crest factor.

A problem we need to tackle in this step is that the Spectral Flux is not scale-independent as the crest factor is. The crest factor has a minimum value of 1 and it can go as high as the difference between peak and RMS of the signal allow. We also know the crest factor value for specific signals. For example, the crest factor of a sine wave is known to be $\sqrt{\mathbf{2}}$. The same things do not hold true for the spectral flux. Spectral flux can vary from a minimum of 0 to a maximum value directly dependent on the number of frequency bins, thus, STFT resolution, we have chosen for the calculation process. Therefore, it is not easy to correspond a SF value to a maximum time constant. In order to overcome this problem we can use the normalized spectral flux and normalize it further to match the high values of the Crest factor. Empirical tests have showed that the crest factor value is usually highest for the very first sample, which is most of the time a global maximum for the function, provides a good normalization factor for the spectral flux.

An alternative could have been, after choosing the number of frequency bins we would use for the STFT, to keep the non-normalized spectral flux values as they were and try to correspond them to a range of time constants using analogies other than the inverse square of the method like we used for the crest factor.

In the next pages, we present comparative figures for the normalized spectral flux and the crest factor for different instruments. For these figures, the time constant($\boldsymbol{\tau}$) used for the peak and the RMS detector in the crest factor calculation (see paragraph 3.3.3) was 50 ms. The spectral flux is presented unmodified, even though we could modify it with a peak detector to get a smooth version of it.

<u>*Figures of Spectral Flux Vs Crest Factor for various Audio Signals*</u>

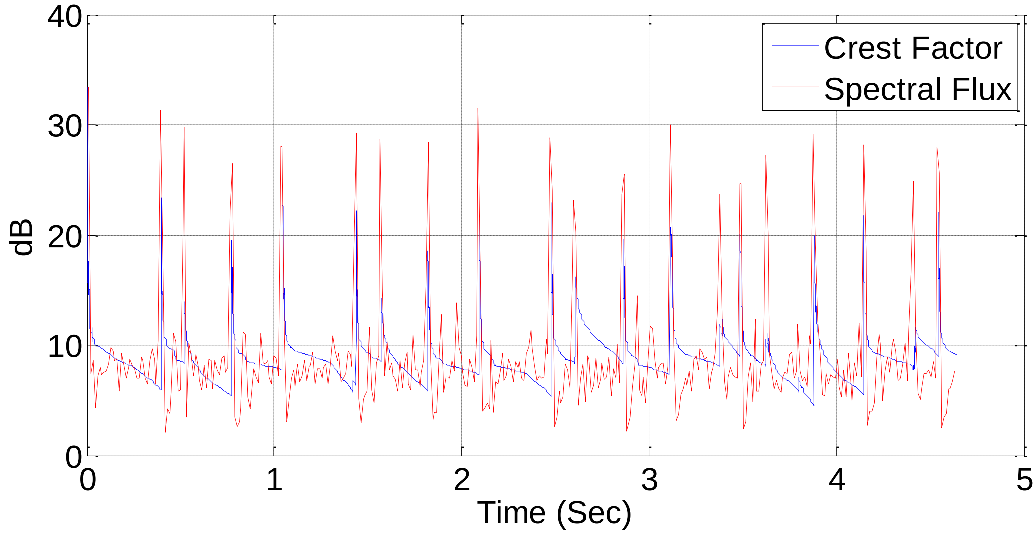


Figure 17. Drums

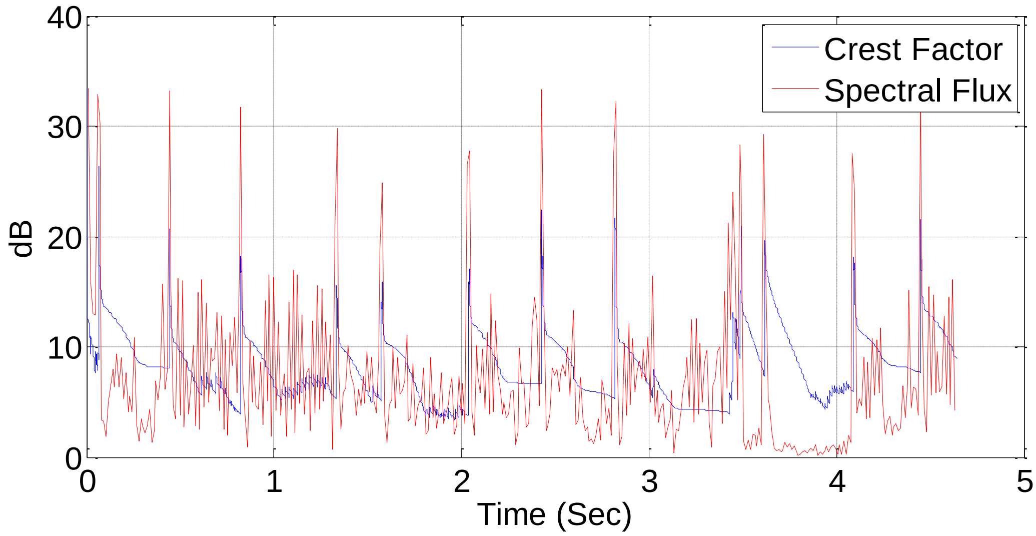


Figure 18. Bass

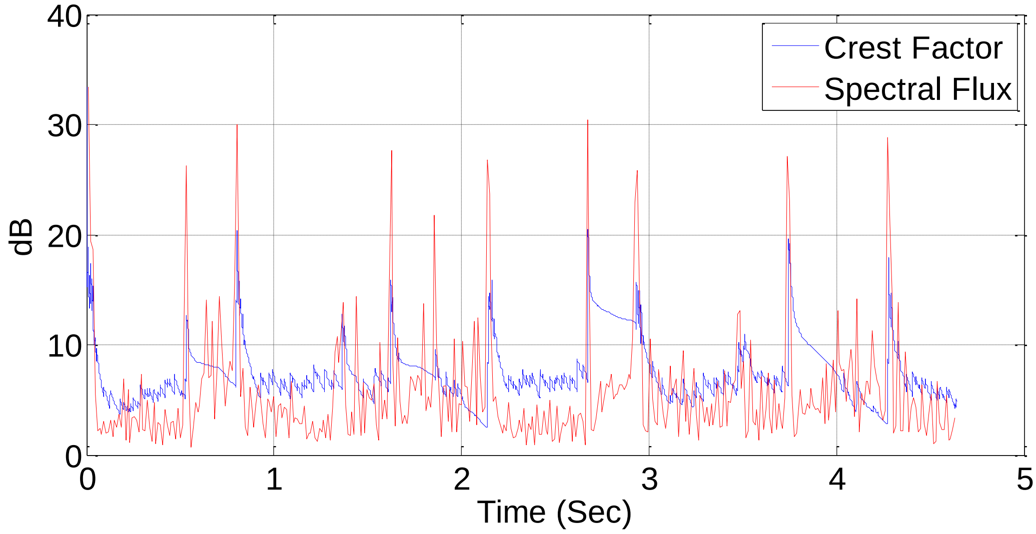


Figure 19. Guitar

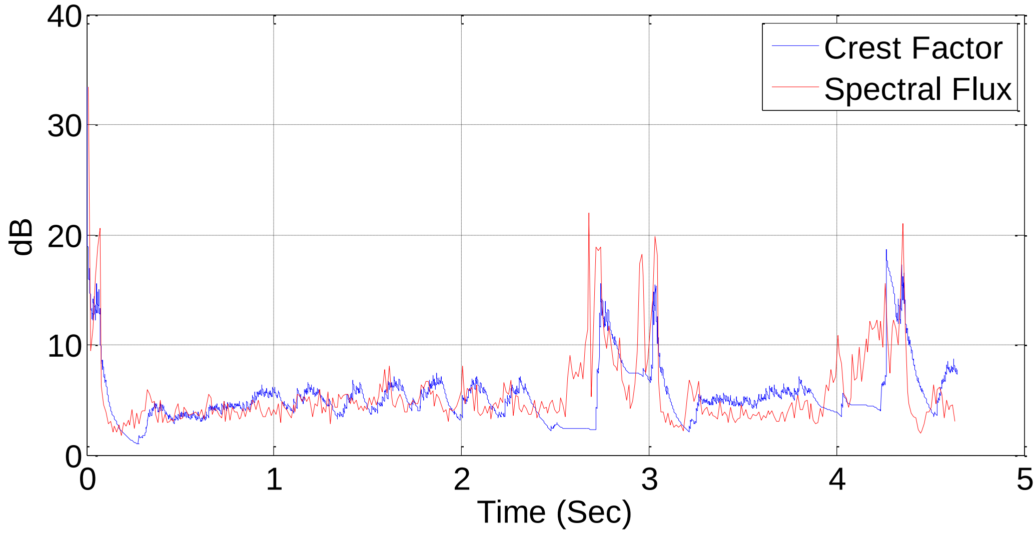


Figure 20. Vocals

After carefully examining the figures we can observe that the spectral flux is very similar in behaviour to the crest factor and equally accurate in the peaks it presents at onsets of notes. Furthermore, it is more "peaky" than the crest factor. This can turn out to be an improvement since it is more sensitive in catching transients. Alternatively, we can smooth it with the use of an appropriate peak detector or a filter if we want a less varying result.

3.3.6 Tackling the Real Time Issue for the Spectral Flux

With the crest factor we were able to obtain a single value per sample and thus, calculate the time constants that correspond to this specific sample. We are unable to do the same with the spectral flux since a window is required for the calculation of the STFT. This will introduce a small latency to the method. Small because the window we chose is 2048 samples long which corresponds to a time length of 46 ms for a sampling frequency of 44100 Hz. The other problem we are facing is that all the sample values inside the window will be assigned a single spectral flux value as opposed to the different crest factor values. We could set the hop size between adjacent frames to be 1 sample and thus calculate a different spectral flux value for every sample but this will increase the computational cost to very high levels (possibly forbidding). Instead we can assign the spectral flux value to all the samples of the window - or in case of overlapping windows, like the 50% overlap we use, in the first half of the samples - and then apply a smoothing filter to gradually smooth out the spectral flux changes as we move from the samples of one window to the other window.

Since this report is not focused on the spectral flux method itself but rather uses it as a tool to obtain some useful results to be used elsewhere (automating compression), we have avoided including detailed results looking at the effect of hop or window size and how these affect the spectral flux function.

3.3.7 Calculating the Time Constants from the Spectral Flux

Figures 16 to 20, present the similarities of the spectral flux and the crest factor. However, Massberg in [2] used a time constant($\boldsymbol{\tau}$) of 200 ms as opposed to one of 50 ms used for these figures after some fine-tuning by listening to samples. As a result the crest factor function he got was a lot smoother but also had lost most of its peak values. We could take a similar approach to that for the spectral flux method and smooth it in a similar way with a peak detector with long attack and release times but we prefer to take a different approach instead. Both time constants, the attack and the release, play a more important role close to the onsets of notes, since its the note onsets that will probably cross the threshold level and trigger the compression. So we want the time constants and especially the attack time to be closely related to the onsets. Therefore, we will try and correlate them to the peaks of the spectral flux which in turn are also closely related to the note onsets. And because the spectral flux peak values are a lot higher compared to the corresponding crest factor values we do not have to use the square of these values like Massberg suggested in [2] to achieve short enough times after transients. In order to keep only the information of the peaks of the spectral flux and also get a function with a similar shape to the crest factor function's shape that Massberg used in his method we will use an instantaneous attack peak detector with a slower release (2 ms). This detector will catch the peaks of the spectral flux and provide a smooth drop after them until the next peak occurs. After a few testing experiments we came up with the following final equations for the time constants:

$$\tau_{Att}[n] = \frac{2\tau_{Att,Maximum}}{y_{SF\ smoothed}[n]} \qquad \tau_{Rel}[n] = \frac{2\tau_{Rel,Maximum}}{{y_{SF\ smoothed}}^{0.8}[n]} - \tau_{Att}[n]$$

We decided to keep the general format of the equations including the same maximum times and the factor of 2. One could experiment more and fine-tune the method further but since the results we get are relatively good and satisfactory there is no need for further modifications.

For the release time we decided to use a power of the SF smaller than 1, since we wanted to achieve longer times. We also used a SF peak detector with even longer release (9 ms) to get less varied release times. The reason is that while we can easily say that the attack times will be activated at the note onsets, we cannot exactly say when the release times will be in use, since this depends solely on when the signal will cross back the threshold level. Therefore, we prefer the release times to vary less after a note onset and not get short very fast. Figures 21 and 22 are two examples where we present the crest factor that Massberg used to obtain his attack and release values as well as the peak detector functions we used to calculate our attack and release times.

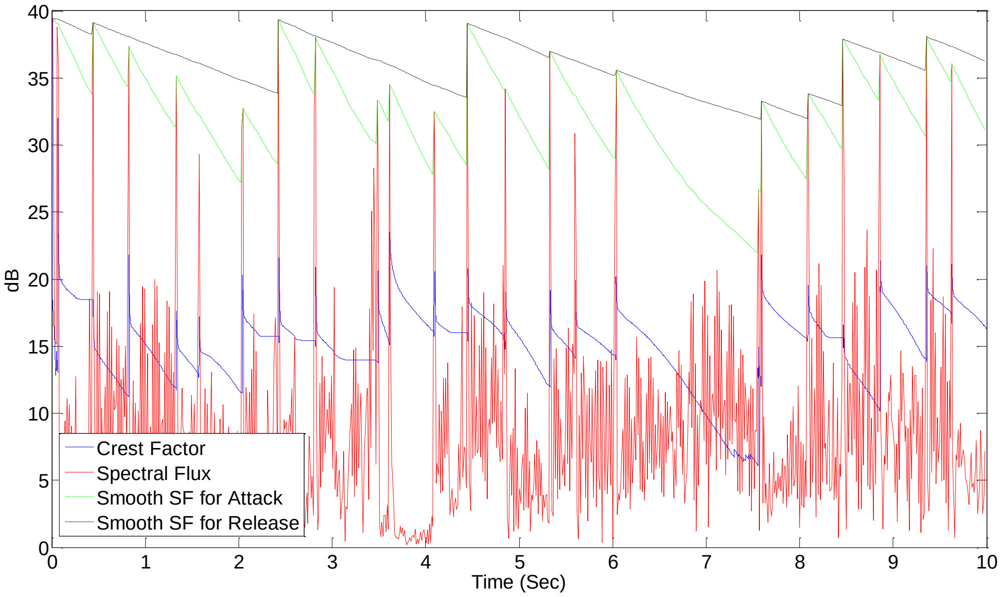


Figure 21. Crest factor and spectral flux method for the calculation of the Attack and Release times (Bass sample)

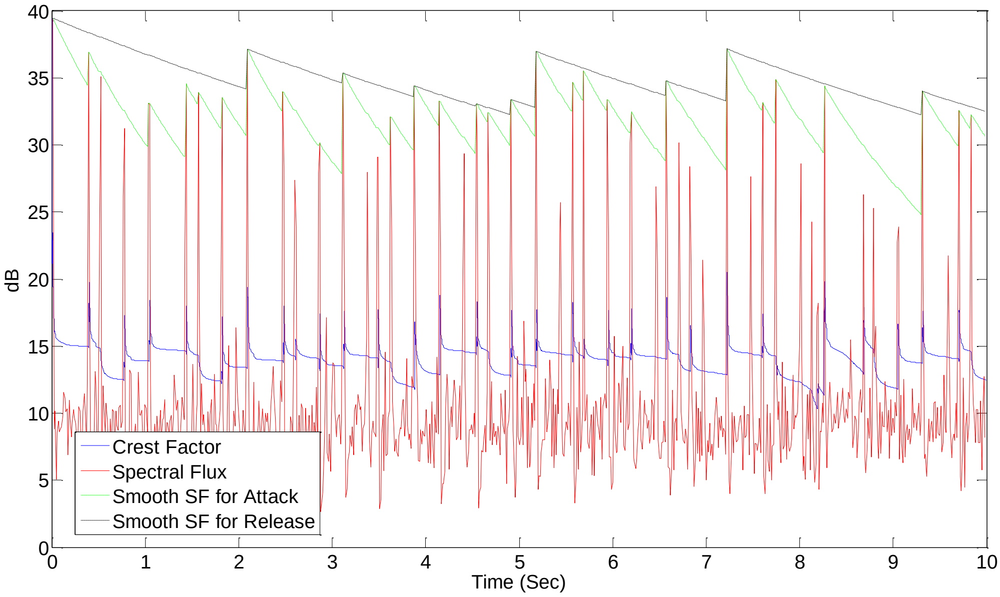


Figure 22. Crest factor and spectral flux method for the calculation of the Attack and Release times (Drums sample)

We should mention here that the method we propose is not necessarily the optimal. One could use the spectral flux itself without the peak detection or with another type of smoothing. The idea behind our choice was to achieve a similar result with that of the method proposed by

Massberg [2] and also use the information of the normalized spectral flux peaks since in our opinion and as explained earlier it is the quite significant for the time constants.

3.4 Auto Make-up Gain

With the make-up gain function we try to equalize the “perceived” volume of the compressed signal to that of the uncompressed signal. Essentially, we try to achieve equal loudness between the compressor input and output signals. The amount of make-up gain needed after compression is exclusively related to the amount of gain reduction applied on the signal by the compressor.

In [1] and [2], there is noted that an auto make-up gain control can be found in some compressor designs. All these implementations calculate the amount of gain required to make the output signal same as the input based on various settings like threshold, ratio and less often, also release. These are all static compensations that are independent of the input signal. Basically, they calculate an estimate of the average gain reduction based on the aforementioned settings and turn this into a set amount of make-up gain. The only way the auto make-up gain will vary is if the compressor settings like the threshold will change. One such example of a gain reduction estimate (control voltage estimate) is :

$$cv_{\mathrm{est}} = \frac{-\log|\mathrm{thr}| \cdot \mathrm{slope}}{2}$$

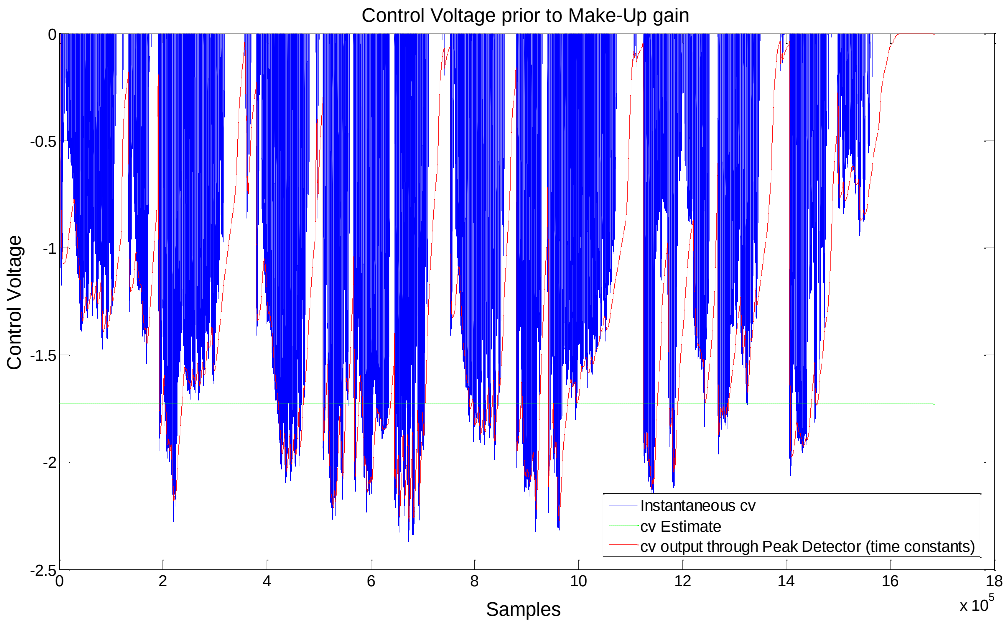


Figure 23. Gain Reduction and Gain Reduction Estimate

3.4.1 The Average Gain Reduction as Make-up Gain

In [2], Massberg proposed an adaptive method for the make-up gain in contrast to the static ones. The method computes the average gain reduction over time and turns it into a varied make-up gain.

We should recall from chapter 3, paragraph 3.1 that the gain computer, responsible for the gain reduction calculation is located inside the log domain of the sidechain. Therefore, we can easily extract the decibel-encoded instantaneous gain reduction per time, since this will be identical to the control voltage of the compressor disregarding scaling. The average gain reduction can easily be considered to be the DC component of the control voltage. So if we regard the average gain reduction as the make-up gain and remove the DC component from the compressor's gain reduction, we will have an adaptive make-up gain control that it is very easily implementable [2].

Removing the DC can be achieved with an averaging low-pass filter. The filter will average the control voltage and following that we will subtract the filter's output from the instantaneous control voltage before applying it to the VCA. See figure 24.

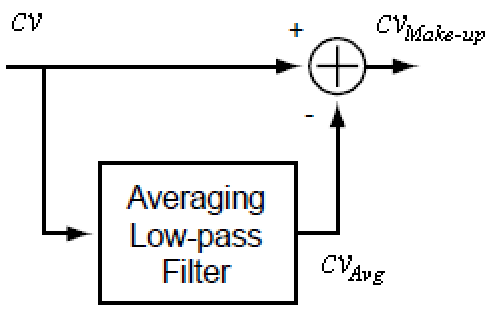


Figure 24. Averaging the control voltage

$$cv_{\mathrm{Avg}}[\mathrm{n}] = \mathrm{cv}_{\mathrm{Avg}}[\mathrm{n}-1] + \alpha_{\mathrm{Avg}}\left(cv[\mathrm{n}] - cv_{\mathrm{Avg}}[\mathrm{n}-1]\right)$$

$$cv_{make-up}[\mathrm{n}] = cv[\mathrm{n}] - cv_{\mathrm{Avg}}[\mathrm{n}]$$

The averaging time constant $\boldsymbol{\alpha}_{\mathbf{Avg}}$ of the filter needs to be carefully chosen. A time constant that is too short, it will follow the compressor's gain reduction (*cv*) to quickly and cancel out the compression. It should not be too long either, because it will then be too slow in following the gain reduction curve and reaching the intended values.

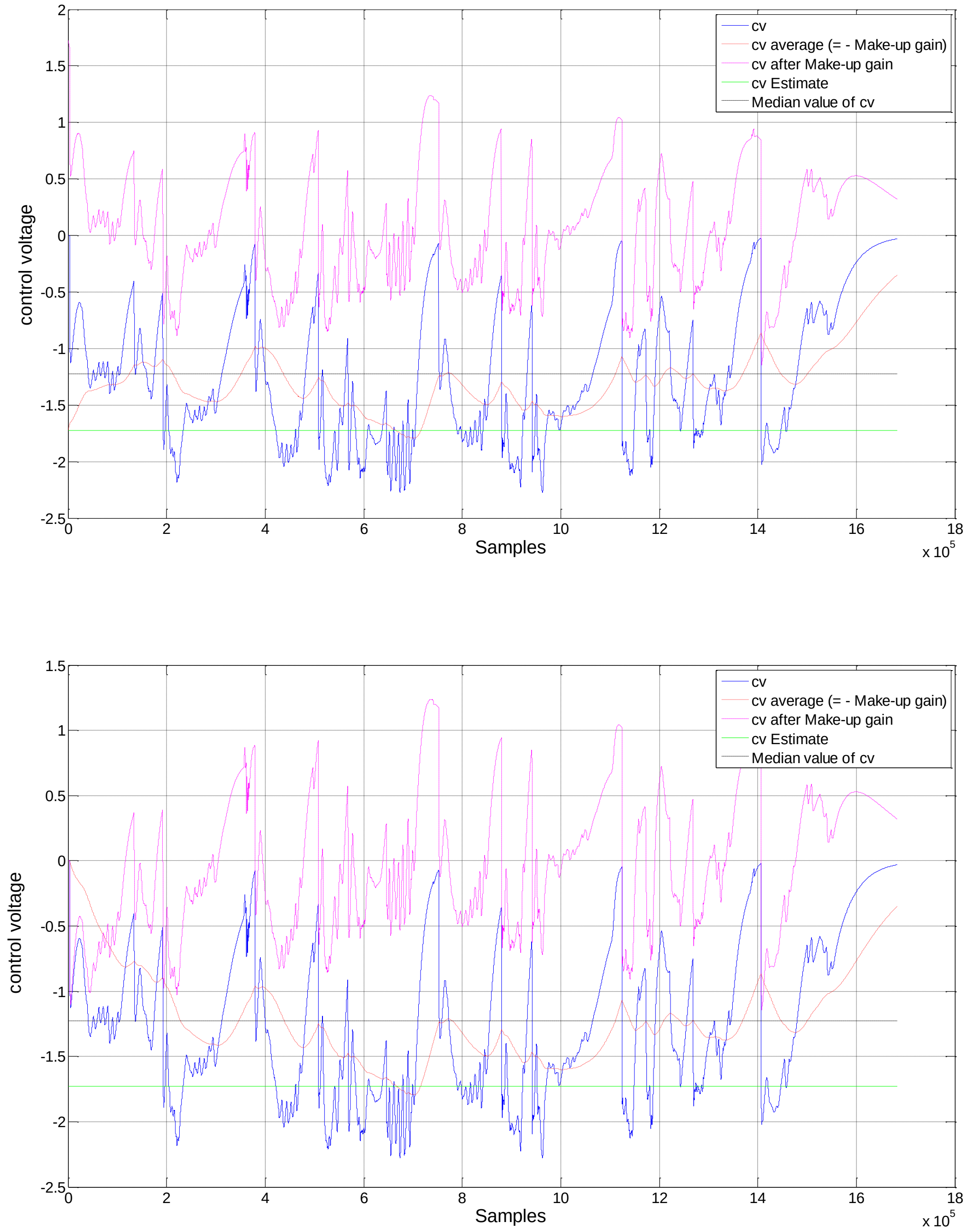


Figure 25. Average Gain Reduction (averaging filter "Hot-started" on the top plot and not "Hot-started on the bottom plot.)

Further improvements for the method, suggested by Massberg in [2], were to use the control voltage estimate to bias the averaging filter, by subtracting the estimate before the filtering and adding it back in afterwards. This method can be described as a ‘Hot-start’ process for the averaging filter since it initializes it at a value that is probably a lot closer to its intended values in the long term. In addition to that, the make-up gain will need a lot less time to adapt to forced parameter changes from the user like a change in ratio or threshold, since the cv estimate will be instantly readjusted to a different value. The bottom plot of figure 25 presents the ‘Hot-start’ function, while the top plot presents the average gain reduction without ‘Hot-start’.

3.4.2 Considerations on the Average Gain Reduction Method

The method presented above is an intuitive one and overall, it performed quite well on various signals as we will see in the results chapter. Nevertheless, the method does not provide a single value in dB as a make-up gain since the average gain reduction is not a set value but a varied quantity. Therefore, the make-up gain introduced by this method will be varied over time instead of a static compensation factor. This alters the compressor’s dynamic behaviour since it changes the overall gain reduction applied on the signal. This can be easily seen with signals that are very varied with sharp changes from quiet to loud parts. The average gain reduction will fluctuate a lot over such a signal and it will not follow perfectly the instantaneous gain reduction. As a result, the make-up gain will vary intensively as well and result in an overall alteration of the gain reduction applied on the signal. In contrast to that, a static make-up gain will just add a DC component on the overall gain reduction of the compressor, leaving its shape unchanged.

For an offline implementation of the make-up gain, the simplest and most effective methodology would be to simply sum up the gain reduction for all the signal samples and divide by the number of samples, thus find the average gain reduction. Then this set value can be used as make-up gain for the compressor. Since this method, analyzes the whole of the signal prior to deciding the make-up gain it is guaranteed to produce fine results.

Other more sophisticated methods could make use of the offline make-up gain method but make use of windows. So they will choose a set make-up gain value to be applied over the samples of one window and a different value over the following window and so on. A smoothing filtering process would be required to level the transition parts between adjacent windows to produce a more smooth make-up gain.

Similarly, for the online implementation with the averaging gain reduction process we could use a look-ahead function (see section 2.7) to give some time to the filter for the average gain reduction to catch up the changes of the control voltage and produce a more effective make-up gain.

3.4.3 Introducing a Loudness measure as part of the Make-up Gain

All the methods mentioned above about the make-up gain do not take into account loudness, even though the main purpose of the make-up gain is to achieve same loudness levels between the input and the output signals in a compressor. Therefore, it would be interesting to see if we can get any better results and improve the make-up gain control through the use of a loudness measure.

3.5 Loudness

Loudness is a subjective term for the auditory sensation of the magnitude of sound as this is perceived by the human auditory system. The American National Standards Institute has defined it as "that attribute of auditory sensation in terms of which sounds can be ordered on a scale extending from quiet to loud." [15]

Human have a limited range of audible frequencies from around 20 Hz to 20 kHz. But the auditory area of the human ear is also bounded by two threshold curves. The threshold of hearing delineating the lowest level sounds the ear can detect and the threshold of pain (or feeling) at the upper extreme of the auditory area. Humans cannot perceive sounds other than those falling inside this area. The area of audibility, as it is presented in figure 26, has two dimensions, the vertical one being the sound pressure level and the horizontal being the range of frequencies the human ear can perceive [16].

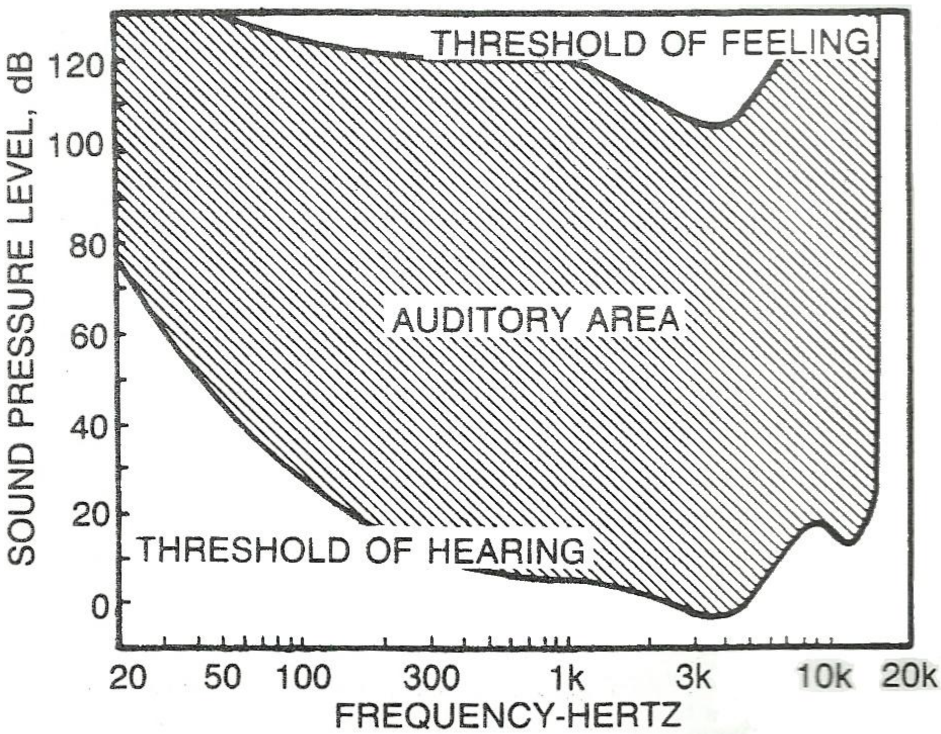


Figure 26. Area of Audibility for the Human Ear. Picture taken from [16]

Surprisingly, human's perceived loudness varies greatly with frequency and sound pressure level. This led to the establishment of a family of equal loudness contours that span the auditory area. Equal loudness contours as presented on figure 27 are the work of Robinson and Dadson back in 1956 [17] and have been recommended as an international standard. It is a revision of the Fletcher – Munson equal loudness contours.

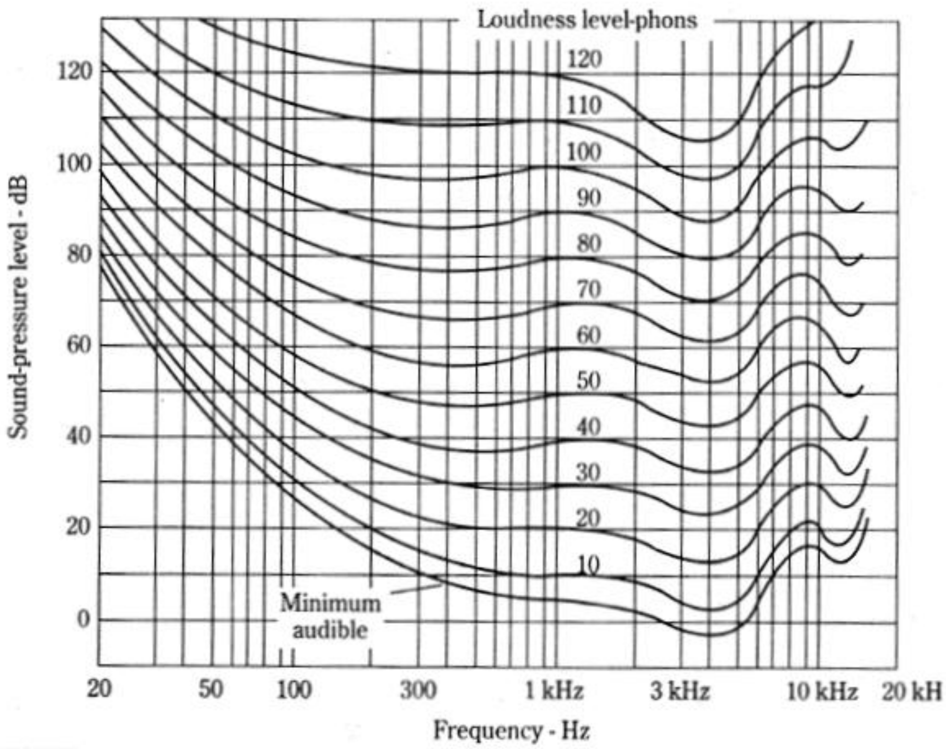


Figure 27. Equal Loudness Contours as defined in [17]

The unit of loudness level is the *phon* and is tied to sound pressure level at 1000 Hz. The curves reveal that two sounds of the same pressure level but of different frequencies can be far from perceived equally loud. This can be seen especially in the low frequencies, where low frequency sounds need a lot more dB in sound pressure level to match, in loudness, a sound of 1 kHz. Based on this interesting loudness feature, we want to test whether by incorporating loudness in our make-up gain model we will end up with better and more accurate results for the auto make-up gain compared to what people would choose as their ideal make-up gain if asked to adjust it manually.

3.5.1 Loudness Measurement

Usually loudness measurements are taken with the use of frequency weighting filters such as A-, B- or C-weighting that attempt to adjust sound measurements to correspond to loudness as perceived by the typical human. However, loudness perception is a far more complex process that just frequency weighting. Frequency weightings approximate the behaviour of the auditory system at different intensities. More specific, A-weighting curve approximates the 40 phon equal loudness contour (relatively quiet sounds), the B-weighting curve approximates the 70 phon contour (medium sound intensity level sounds) and C-weighting curve approximates the 100 phon contour (loud sounds). There are also additional curves as well as variations of the ones explained here. It should be noted that frequency weighting curves are just simplified approximates of the equal loudness contours. The purpose of the filtering with the weighting curves is to take objective measures of the perceived loudness of audio signals. Having objective measures of loudness is desirable in numerous applications like broadcasting. Objective loudness measures are simply frequency-weighted versions of an *Equivalent Sound Level* (Leq) measure [18]. Leq is defined as:

$$\mathrm{Leq(w)} = 10\log_{10}\left[\frac{1}{\mathrm{T}}\int_{0}^{\mathrm{T}} \frac{\mathrm{x_w}^2}{\mathrm{x_{ref}}^2}\,\mathrm{dt}\right], \mathrm{dB}$$

Where w refers to a specific weighting filter from A, B, C etc. $\boldsymbol{x_w}$ is the audio signal at the output of the weighting filter $\mathbf{w}$, and $\boldsymbol{x_{ref}}$ is some reference level and T is the duration of the signal.

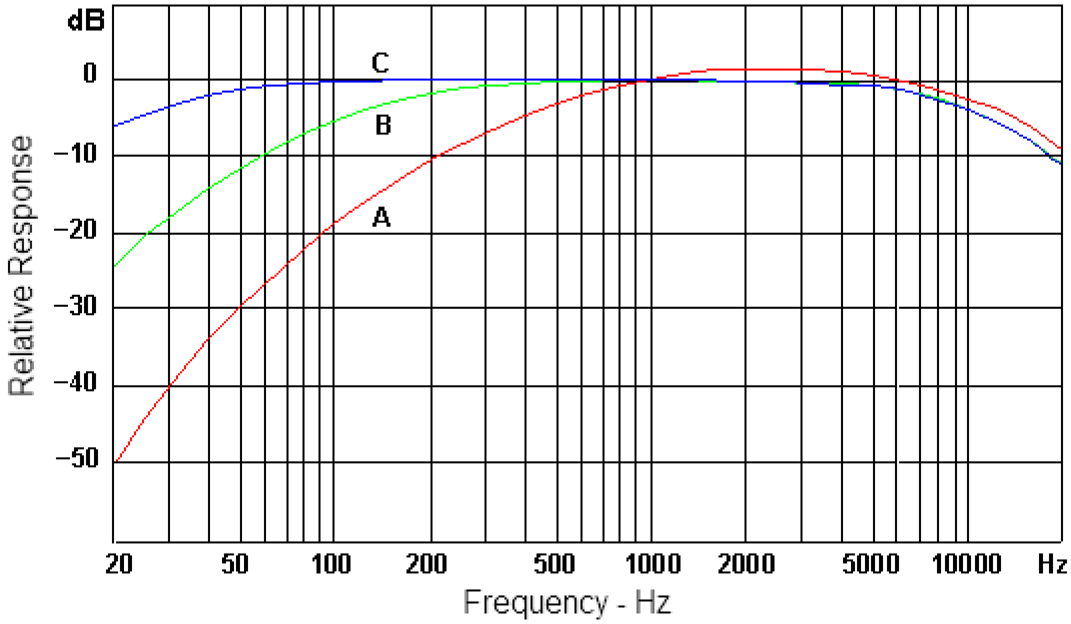


Figure 28. Frequency Weighting Filters

In [18], several basic objective measures of loudness were evaluated based on their ability to predict the relative loudness of monophonic program material reproduced over a single loudspeaker. The results of that evaluation indicated that the Leq(RLB), a Revised Low-frequency B curve remaining flat on the high end (see figure 29), was the best of the basic objective loudness measures, followed closely by Leq(C). This is in conflict somewhat with the findings of Aarts [19], who examined the problem from a different perspective; that of matching the loudness of different loudspeakers. Aarts found that a B-weighted loudness measurement outperformed the C-weighted one. However he used a different evaluation metric than the one Soulodre and Norcross used in [18].

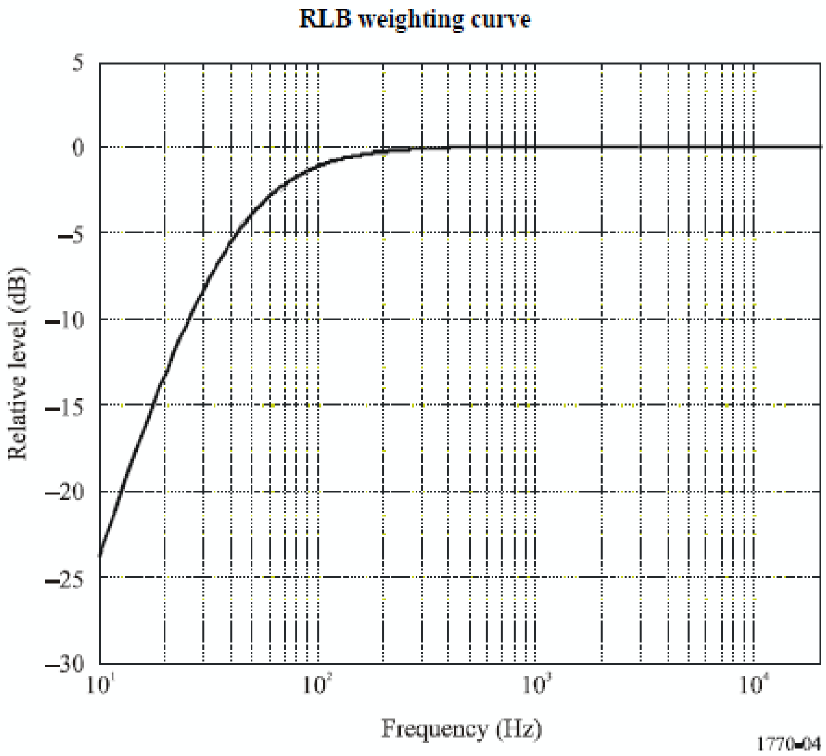


Figure 29. The RLB frequency weighting filter magnitude response

Furthermore, more elaborate models for measuring loudness can be found in the literature. These methods are based on psychoacoustical data of human ears and can be used to measure absolute loudness, in contrast to the weighting curve measures that are only intended for rank ordering of sounds – noises according to loudness [19]. Such models are the Zwicker loudness model, which became the ISO 532B standard [20], and another model proposed by Glasberg and Moore [21], expanded later [22] to overcome certain limitations. These are far more complex models and because of that they have an increased computational cost that may prohibit them from being used in some applications. In addition to that, all the loudness level estimates show a high correlation with each other, and if someone makes the assumption that psychoacoustic loudness models like that of Zwicker or Moore represent the real perceived loudness (absolute loudness), then the error produced by using average A-weighted levels is about 2.5 phons [23]. At the moment, the field of loudness measurement is a very active one. There is ongoing research that has produced some fruitful results like a new objective measure of loudness [24] or a gated loudness measurement [25].

3.5.2 A Loudness Measure for the make-up Gain

There is no reference in the literature about an attempt to introduce a loudness measure in the make-up gain stage of a compressor, or at least no one that is known to the author of this paper. R. J. Cassidy in [26] designed a dynamic range compressor whose steady-state and time-varying level detection characteristics match the loudness characteristic of the human auditory system. A typical compressor derives a sound level that is independent of frequency when the input is a steady state sinusoid. However, the human auditory system is not equally sensitive to steady state sinusoids but rather frequency dependent. Cassidy designed a filter with a varied magnitude frequency response that approximated the equal loudness contours. The side chain signal was passed through this filter and therefore the gain reduction that the compressor applied to the signal was calculated based on its loudness measurement rather than the signal itself. Although this method could provide improved dynamic range reduction for a variety of signals, it alters the basic principles behind the compression effect and follows a different direction than the one we have set for this project.

In our compressor design, we want to use a loudness measurement to measure and compare the perceived loudness between the uncompressed and the compressed signal. By this direct comparison we will be able to extract the loudness difference between the two signals and use it to calculate the make-up gain needed for the compressed signal in order to match the loudness levels if the uncompressed signal. After obtaining the results we will be able to tell whether the loudness model provides anything beyond a simple gain reduction averaging process; so whether it can be considered as an optimization for the make-up gain stage or not.

Initially, we used objective loudness measures like B- and C-weighting to measure the loudness of the signal in the input and the output of the compressor. For the use of these measures we had to realise the characteristics of the weighting contours. This was done by using the definitions of the analogue weighting filters taken from the s-domain transfer functions of the weighting contours, explained in [27] and also described in ANSI Standards S1.4-1983 and S1.42-2001. From the analogue filter and using the bilinear transform in Matlab we were able to extract the filter coefficients of the digital weighting filters. The digital weighting filters were used to produce loudness measurements for our compressor implementation.

The results revealed that the method was inappropriate for our intended purposes. The audio signal prior and after compression has the same frequency content and the only thing that is altered by the compressor is its dynamic range. Filtering the input and the output with the same weighted filter, will result in the same alterations (magnitude changes) introduced on both signals and the only difference between the two will be that in their dynamic range introduced by the compressor. Therefore, by directly comparing the loudness of the two signals we will end up with a dB difference that is the result of the compression being applied on the signal, rather than any other changes introduced because of loudness. To express it in other words, the filtering process is the equivalent of convolution in the time domain, but this corresponds to multiplication in the frequency domain. So for example let $\boldsymbol{X}$ be the signal before compression and $\boldsymbol{Y}$ after compression, then from section 2.2 the gain reduction $\boldsymbol{G}$ in an ideal VCA is equal to the exponent of the control voltage:

$$\boldsymbol{G}[\boldsymbol{n}] = \mathbf{exp}(\boldsymbol{cv}[\boldsymbol{n}]) = \frac{\boldsymbol{Y}[\boldsymbol{n}]}{\boldsymbol{X}[\boldsymbol{n}]}$$

When dealing with compressors, G always refers to gain reduction since the control voltage ($\boldsymbol{cv}$) is always negatve (output is attenuated compared to the input). In expanders the control voltage is positive and this results in amplification of the output compared to the input, thus a gain increase.

Massberg's Make-up gain $\boldsymbol{G_{make-up}}$ was the average gain reduction. This was achieved with a low-pass filter $\boldsymbol{H_{low-pass}}$, so:

$$\boldsymbol{G_{make-up}}[n] = -H_{low-pass}(G[n]) = -H_{low-pass}\left(\frac{Y[n]}{X[n]}\right)$$

The loudness-based make-up gain introduces a filtering process of the input and output with a weighting loudness filter $\boldsymbol{H_L}$. So:

$$\boldsymbol{G_{make-up}}[n] = -H_{low-pass}\left(\frac{\boldsymbol{H_L}(Y[n])}{\boldsymbol{H_L}(X[n])}\right)$$

But $\frac{\boldsymbol{H_L}(\boldsymbol{Y}[\boldsymbol{n}])}{\boldsymbol{H_L}(\boldsymbol{X}[\boldsymbol{n}])} = \frac{\boldsymbol{Y}[\boldsymbol{n}]}{\boldsymbol{X}[\boldsymbol{n}]}$ so we end up with the same make-up gain as Massberg's method since, filtering both the input and the output with the same filter will result in the filtering being cancelled out from the nominator and the denominator and we will end up on just the output/input ratio which is simply the compression. What is needed for the method to work is the use of multiple weighted filters that will correspond to the various equal-loudness contours as these were defined in [17].

3.5.3 A Model with Multiple Equal Loudness Contours for the Loudness Measure

The model we will use will contain the ISO 226 Standard Loudness Contours as this has been used in [28]. The model consists of a look-up table with all the filter coefficients necessary to depict the equal loudness weighting curves. Implementing a series of weighted filters representing the behaviour of the equal loudness contours is a very laborious task. The contours are non-linear and not monotonically increasing or decreasing for the most part. This results in a complex process for estimating the filter coefficients and in all probability in filters of very high orders that will introduce latency to the method and be computationally quite expensive to implement. Therefore, it is in our best interest to simply approximate the behaviour of the equal loudness contours with a series of simple, easily implementable filters, somewhat similar to the weighting filters used in the objective loudness measurements.

To approximate the behaviour of the equal loudness contours from the lower frequencies of the auditory area up to around 10-12 kHz, which defines a range that includes most of the energy of music and audio signals, we need 4 filters. These can be digital biquad filters. A digital biquad filter is implemented as a second order recursive linear filter, containing two poles and two zeros. It's equation is the following:

$$y[n] = b0 * x[n] + b1 * x[n-1] + b2 * x[n-2] - a1 * y[n-1] - a2 * y[n-2]$$

In figure 30, we present the block diagram for a digital biquad filter in its 1st direct form.

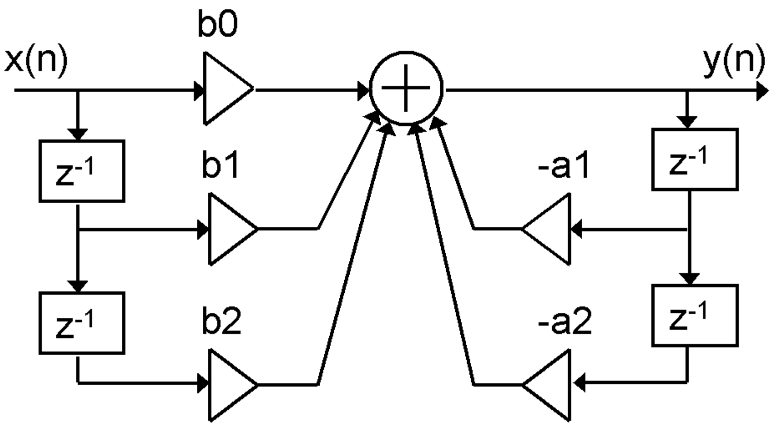


Figure 30. Digital Biquad Filter (Direct Form 1)

The first of the 4 filters that we need, will be a low-pass filter approximating the decreased sensitivity of the human auditory system to low-frequency sounds. The second filter will be a soft notch filter (inversed peak filter), approximating the local maximum that the equal loudness contours introduce around 1 kHz. Similarly, the third filter will be peak filter approximating the local minimum of the contours and the increased sensitivity of the ear around 3-4 kHz. Finally, the last filter will be a very wide notch filter approximating the decreased sensitivity of the ear at the very high frequencies. The filters will be connected in series and the signal will pass through all of them one by one. In figure 31, we present an example of a set of 4 filters used to approximate the behaviour of the 80 phons equal loudness contour.

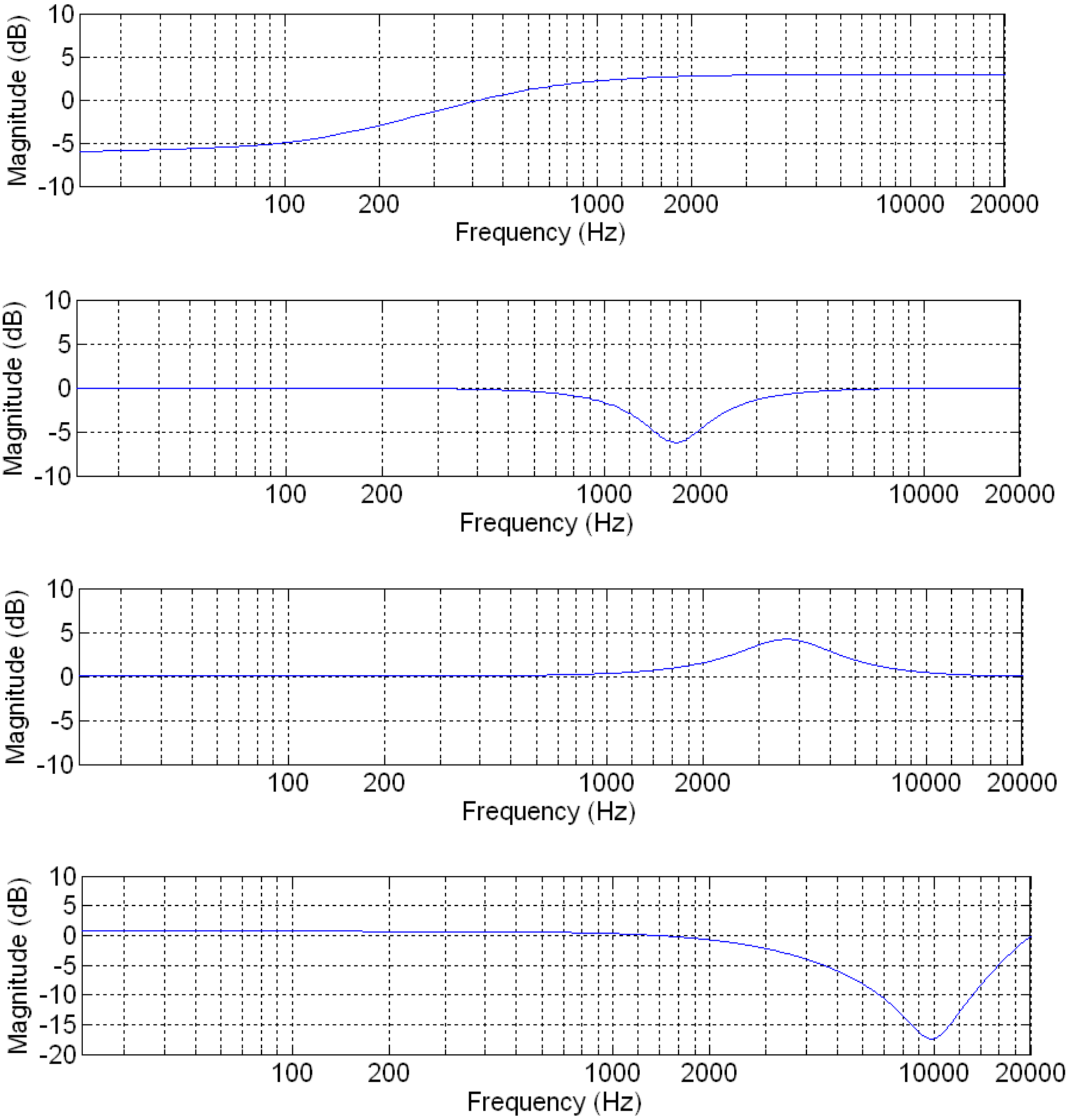


Figure 31. An example of the 4 biquad filters magnitude response for approximating the Equal Loudness Contour of 80 phons.

After building the filter bank for the equal loudness contours, there remains one thing left to implement. That is, a way to measure the sound pressure level of the input signal. Each one of the various filters, the design process of which we explained above, corresponds to a specific range of SPL. For instance, if the SPL is between 95 and 105 then the corresponding weighting filter to be used is the one related to the 100 phons equal loudness contour, if the SPL is between 85 and 95 then the filter to be used will be the 90 phons equal loudness contour weighting filter and so on. Therefore, we need to be able to correspond input signals, or even better sample values, to corresponding sound pressure levels.

In Matlab, sounds read from WAV files are normalize to have an amplitude range between -1 and 1. In this process, any information about the true recording or playback levels of the sound is lost. If we knew the approximate SPL or if we have a desired value of dB SPL, we could calibrate the signal using a reference level (usually the threshold of hearing – 20μpa) and afterwards scale it to match the desired value of dB SPL [29]. Since rescaling the signal will alter the amplitude range of its values and will require a readjustment of the compressor parameters to match the new scale of the signal in order to have the effect work properly a simpler method has been used.

We can calibrate the signals so as the maximum possible value in Matlab, 1, will correspond to an SPL level of 125 dB which corresponds to the topmost equal loudness contour. Samples with values smaller than 1, will be directed to dB SPL values smaller than 125 based on the following equation:

$$\mathrm{SPL}[n] = 10 * \log_{10}\left(\frac{x[n]^2}{10^{-\frac{125}{10}}}\right)$$

This method does not calculate the correct SPL levels of signals, but this does not come into conflict with the purpose we are using it. We do not want to measure exact SPL levels but rather to be able and compare the differences between SPL of the compressed and uncompressed signal sample values. Apart from the initial assumption that a Matlab value of 1 corresponds to a maximum SPL of 125 dB and that all the audio samples had similar recording and playback levels, the method does not deviate from what it is expected from theory. So for a 16 bit soundfile, the maximum value is 32767 which corresponds to an amplitude of 1. Every successive halving of amplitude corresponds to -6 dB in level [30]. This can be taken as rule of thumb and the equation we present above satisfies this rule since if a single of amplitude 1 has a corresponding SPL of 125 dB, a signal of amplitude 0.5 will correspond to approximately 119 dB of sound pressure level.

3.5.4 Testing the Loudness Measure

Having built up the loudness measure model with the equal loudness curves filter bank and the SPL measurement, one thing remaining is to test how it performs under simple conditions. For that reason we used some audio samples and applied heavy compression on them. For the heavy compression we used similar settings as the ones used by Massberg in [2] for his make-up gain evaluation experiment. These were: threshold at -30 dB, ratio at ∞:1, knee width at 0 dB (hard knee), attack time at 0.5 ms and release time at 100 ms. An example of one of the samples before and after compression is presented in figure 32. Afterwards, we computed the loudness measurements for the samples before and after being compressed and made a direct comparison between the difference in the signal's amplitude and the signal's loudness before and after compression. The results can be seen in figure 33.

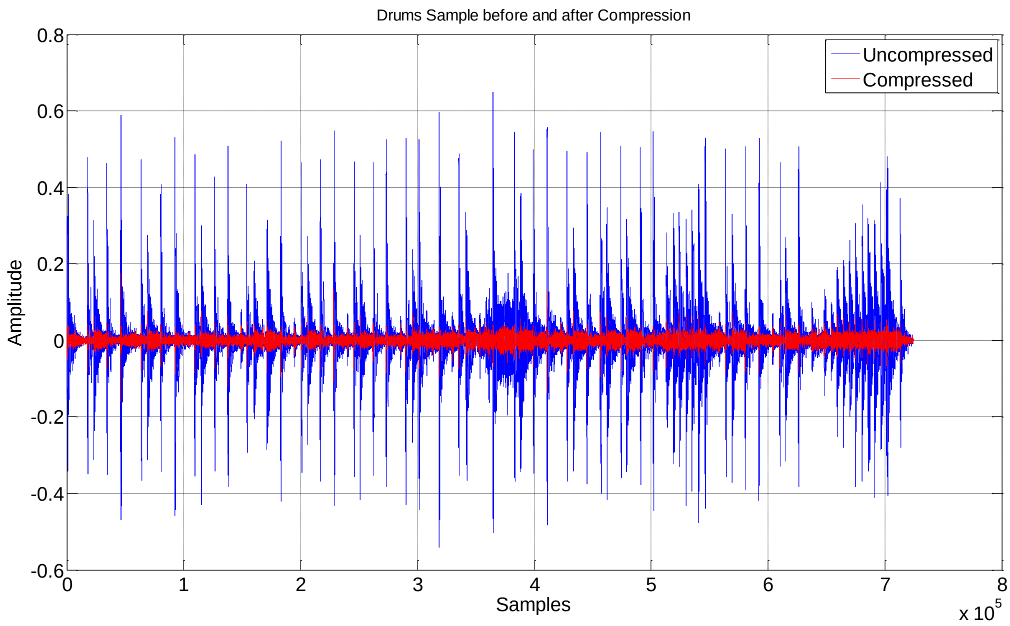


Figure 32. Drums sample before and after being compressed

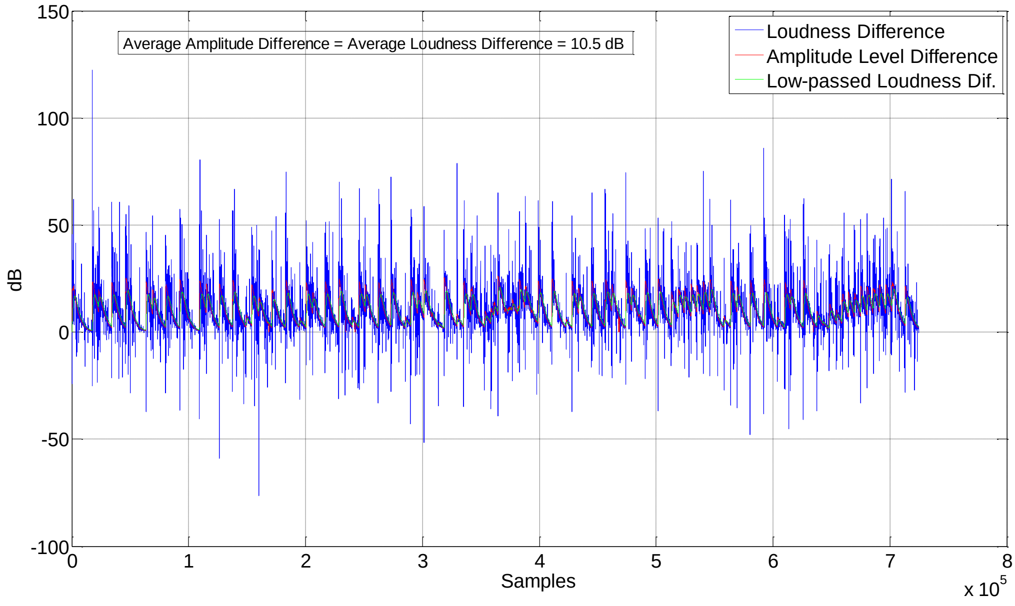


Figure 33. Difference in dB between the Amplitude and the Loudness of the uncompressed and the compressed Drums sample

Whether we did a direct comparison of amplitude differences and loudness differences per sample or we calculated the mean values in each case and compared them the result was that the loudness measure did not change the dB difference between the compressed and the uncompressed signal significantly. In most cases the difference was less than 1 dB. Therefore, we cannot depend on the loudness model to improve the make-up gain control as we expected that it would do.

Even a more sophisticated idea, involving extracting the probability density function (pdf) of the Loudness measure for a signal before compression and after compression and computing the difference between the maxima (most probable values) between the two, did not provide more than a couple dB of increase in the make-up gain computation at best and thus, was also considered insufficient as an improvement for the method. In figures 34 and 35, we present the pdf of loudness for the uncompressed and the compressed signal respectively.

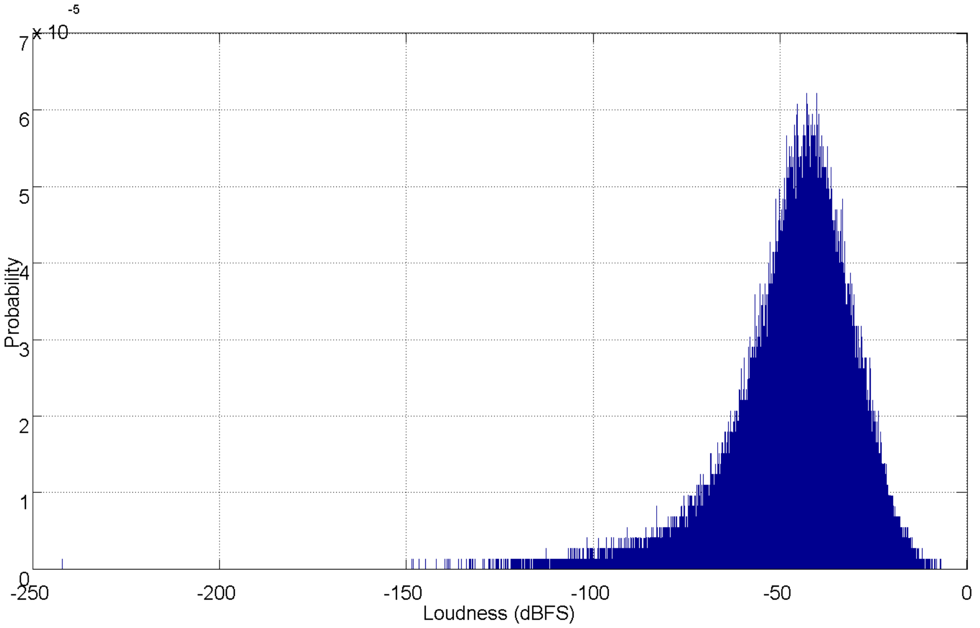


Figure 34. Probability Density Function of loudness for the uncompressed Drums sample

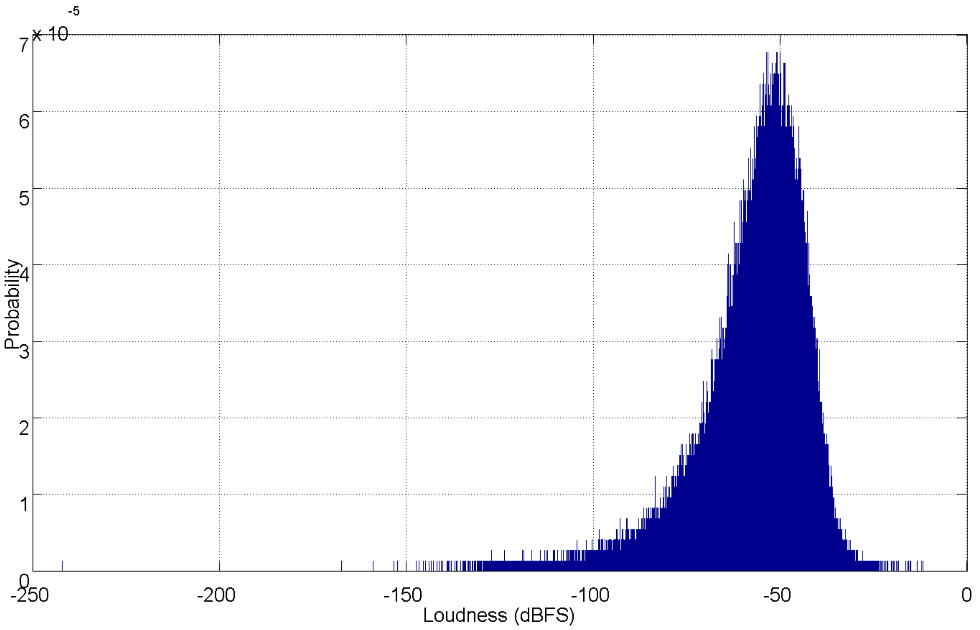


Figure 35. Probability Density Function of loudness for the compressed Drums sample

3.5.5 A Time-based Approach in the Perception of Loudness

The solution to improving the make-up gain result might still be hiding in the perception of sound by the human auditory system but not in loudness. Instead it might be in the duration of sounds and the different levels of sensitivity our ears exhibit to sounds of different lengths. Both drums and slap bass signal samples are characterized by very high peaks with very short time duration and with a large transient content located mainly in those short length peaks. When those peaks are limited by the compressor and these transients are reduced in amplitude, this has a significant impact on the perception of loudness. The explanation for that is that our ears are less sensitive to short transient sounds [16]. Short sounds must be louder in order to be comparable to longer sounds as it can be seen in figure 36. Therefore, sounds with duration shorter than 200 ms, like those early transients in the onset of drums and sample bass signals, will be less audible and have an important impact on loudness when they are reduced in SPL because of the compressor's gain reduction process.

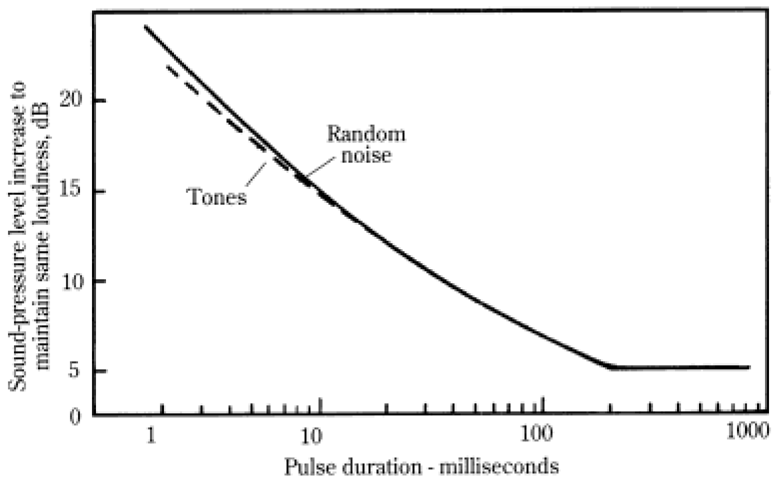


Figure 36. Short pulses of tones or noise are less audible as this graph illustrates. The discontinuity in the 200ms region is related to the integrating time of the ear. (Taken by [16])

It might be interesting to calculate the mean duration of the peaks of the signal that go above the threshold, thus the notches in the control voltage, and then check whether this duration is short enough so as to allow for a boost in the make-up gain in order to compensate for clipping these short in time peaks and thus making them a lot less loud compared to the rest of the signal. Of course, measuring the duration of the signal parts that are to be compressed can be tricky. In order to check the duration of the control voltage peaks we need to make use of a threshold value probably other than the compression threshold since if that is set too low, like in cases where heavy compression is applied on the signal, it might be applying a gain reduction over a continuous part of the signal and not just the peaks. This does not mean though that the signal is not “peaky” and that there is no need for an increase in the make-up gain to make the compressed signal perceived as equally loud to the uncompressed. The role of this varied, dependent on the compression amount, threshold can be fulfilled by the average control voltage. Although this is not a constant value but a varied one, it follows closely the control voltage and adapts itself to the changes of it so it is always crossing the notches of the control voltage at a certain level neither to high nor to low. We are only interested in the average time length of the control voltage notches (which directly correspond to the signal’s peaks) so we can allow for a small error (deviation from the correct value).

After calculating the average time duration of the signal’s compressed peaks we need to correspond that to a make-up gain boost. That boost has to be varied depended on the signal’s compression amount (or threshold value). So figure 36, it cannot be of much help here. A clever way is to use the mean value of the average control voltage, as presented earlier, which is directly dependent on the amount of compression. More accurately, we will use a percentage of the that value, that we can be sure that corresponds to a sufficient make-up gain boost from the evaluation test in [1]. So for the amount of extra dB to be added on the signal we come up with the following equation:

$$Additional\ Make-up\ Gain = \begin{cases} -q \cdot \overline{cv_{Avg}} & if\ D > 200\ ms \\ 0 & if\ D \geq 200\ ms \end{cases}$$

Where D stands for average time duration of compressed peaks, $\overline{\boldsymbol{cv_{Avg}}}$ is the mean value of the average gain reduction and q is a scaling constant for the control voltage estimate which was fine tuned at a value of 0.3. The minus sign denotes that this corresponds to an increase in the gain and not a decrease.

We would like to be able and develop a more sophisticated method where the additional dB of increase for the make-up gain will be directly related to the time length of the compressed peaks but the inability, due to insufficient time, to perform an accurate time length measurement and more importantly the inability to test that measurement and its performance on various audio samples, forced us to present a basic, yet efficient, method to improve the make-up gain.

3.6 Threshold & Ratio in the Auto Compressor

Both the threshold and the ratio parameters relate to the static compression characteristics (refer to the input/output curves in figure 3 and 4). For an auto compressor, we want the user to have to adjust only a single setting that will define the desired compression amount they want to apply to the audio signal. This can be achieved either by automating the ratio and leaving the threshold as the user adjustable parameter or vice versa.

Massberg, in his project [2], left the threshold to be manually chosen, set the ratio parameter to infinity and automated the knee in such a way that the knee width will vary with time depending on the compression of the signal. The idea was that a varied soft knee can be seen as an automatic ratio. We will speak more of this method in the Auto Knee paragraph of this report.

Another implementation could be to let the ratio be manually adjusted by the user and have the threshold automated. An interesting idea for automating the threshold is to have it follow the RMS value of the signal and more correctly a multiple of the RMS value. For example, the RMS value of a sine wave ($\boldsymbol{y = A\sin(2\pi f t)}$) is $\boldsymbol{A/\sqrt{2}}$. If we set the threshold to follow the $\boldsymbol{\sqrt{2}}$*RMS then it will follow the peaks of a sine wave but with the slowly varied characteristics of the RMS. Transients will exceed the threshold value and will trigger the compressor, while steady state sinusoids will not. The method of setting the threshold at the RMS value at any given time has been already tested with success by the author in [31].

Both methods do not alter the compression effect like attack and release times do. They simply affect the compression amount. So it is hard to test them and say whether they perform satisfactory or not, unless we present audio files as part of a listening test to a big group of subjects and ask them to decide whether the amount of compression in the audio files is satisfactory and also rank the various audio files. Because of the limited time we have at hand to finish this project, such a test, properly organized to avoid misleading results, is impossible to be performed.

3.7 Auto Knee

Knee is probably the hardest parameter to automate since the choice of it depends highly on the user's intentions and personal taste. Both hard or soft knee can be used on a signal depending on what one wants to achieve by compressing that signal.

Massberg, in [2], created a soft knee with a varied automated width based on the idea that such a knee can be seen as an automatic ratio. In the soft knee implementation and in the dB range of the soft knee we can consider the ratio to be equal to the instantaneous slope of the knee. If we differentiate the gain computer equation (see paragraph 2.4.2) we can get the instantaneous slope of the output of the logarithmic gain computer.

$$\mathbf{slope_{Inst}(x) = f'(x - log|thr|) \times slope}$$

where the derivative of the soft rectifier function is:

$$f'(x) = \begin{cases} 0 & \text{for } \; x < -\log|\mathrm{width}|/2 \\ x + \dfrac{\log|\mathrm{width}|/2}{\log|\mathrm{width}|} & \text{for } \; -\log|\mathrm{width}|/2 \leq x < \log|\mathrm{width}|/2 \\ x & \text{for } \; x \geq \log|\mathrm{width}|/2 \end{cases}$$

Now if we recall from section 2.4.1., the rectifier's input is the logarithm of the input signal minus the logarithm of the threshold or *log|in|-log|thr|*. For a specific ratio of ∞:1, a signal will be perfectly limited as it exceeds the threshold value and will be left unchanged (ratio 1:1) if its level is lower than the threshold.

If we substitute the rectifier with a smooth one and keep the ratio ∞:1 then, the signal will be perfectly limited once it exceeds *log|thr|+log|width|/2*. Below that point the ratio will gradually decrease, reaching 2:1 exactly at *log|thr|* and it will keep decreasing until *log|thr|-log|width|/2* where it will become 1:1 (no compression at all). So in [2] it is claimed that simply by setting the ratio to infinity and just by turning the knee width one can access the whole range of compression ratios. This concept but without varying the knee width has also been used before [6].

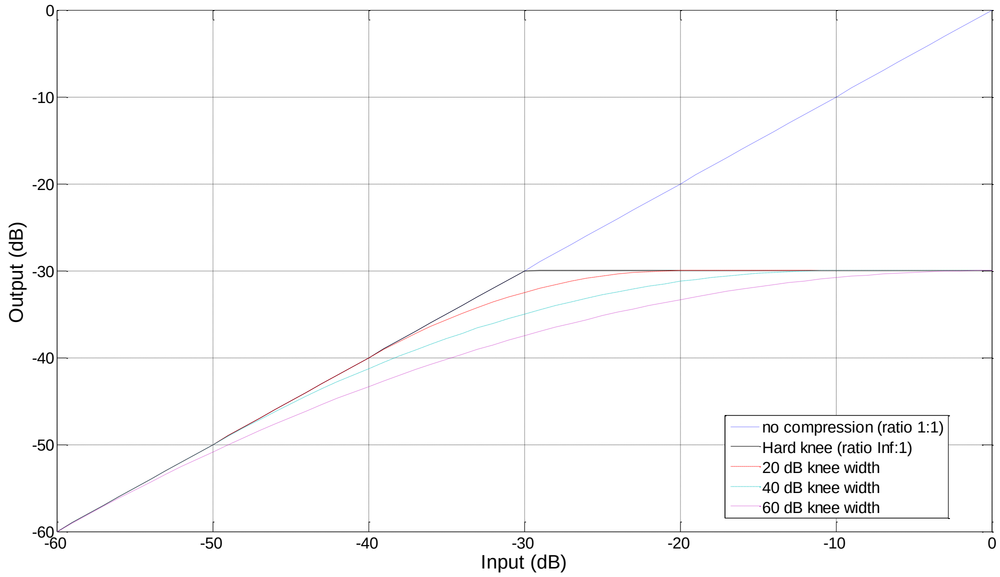


Figure 36. Compression Input/output curves with various knee widths for a set threshold at -30 dB

3.7.1 Automating the Knee Width

In [2], Massberg designed an adaptive method for automating the knee width. The method is based in the simple assumption that if the compression applied on the signal is for short periods of time, so only a few peaks are trimmed and the average gain reduction is small then one might want the compressor to act as a hard limiter for more efficient results. On the other hand, if the signal is heavily compressed, so the gain reduction is constant for the whole signal duration and the average gain reduction is high, one might want a smoother and less obvious compression effect. The use of a soft knee with a large knee width will give this soft characteristic for the most part, whilst short peaks with very high values will be still compressed with a ratio of ∞:1 as from a hard limiter. This is achieved simply by using the average control voltage calculated already for the make-up gain control and multiplying it with a constant value *m* for scaling purposes. The result is a slowly and smoothly varied with time $log|width|$.

$$\mathrm{log|width|[n]} = \mathrm{m} \cdot cv_{\mathrm{Avg}}[\mathrm{n}]$$

The scale factor m of the method was fine tuned at a value of 2.5 after listening tests.

The drawback of the method is that it is exclusively related to the average gain reduction, or in other words to the compression amount applied on a signal, and not at all at characteristics and information of that signal. If the signal is slightly compressed the knee width will be small. If the signal is compressed more extremely then the width will be a lot longer.

3.7.2 Altering the Method to Include Information on the Input Signal

We come to suggest an altered version of the method that will include signal information. More specifically, we will use the spectral flux again as a method to extract some information on the input signal. The idea is that signals with extensive transient content, like drums, will have their spectral flux values above a certain threshold, considerably higher compared to that of a signal with fewer transients, like guitar. For example the spectral flux values of the drums sample, in figure 36, are constantly above 0.1 and around 0.2 while the spectral flux values of the guitar sample, in figure 38, are around 0.05. If we extract the average level of the minimum spectral flux values of a signal and add this information to the knee width calculation, then we would have successfully automate it in a way to include extra adaptability based on the signal characteristics.

We will need to use the normalized spectral flux we computed to use for the time constants. We need to calculate the average levels of the minima of the spectral flux. This can be achieved by first measuring the peak values of the inversed spectral flux. For this we will use a peak detector as this was defined in [2] by Massberg and present the negative values of the spectral flux in its input. That way the minima of the spectral flux will be the smallest negative values, thus appear as peaks. Having calculated the minima we will use a low-pass filter to find the average of these values. This will produce a smooth running average which will differ from one instrument to another. Both detectors were fine tuned to perform in a desired way.

Next step is to adjust the auto knee width function defined above in a way to include this information on the signal extracted from the spectral flux. The method will be still based on the average gain reduction, thus the amount of compression, but it will be scaled with the information from the spectral flux. The logarithm of the width of the knee will increase with the control voltage but in an adaptive way, based on the spectral flux minima of the signal. If the signal has constantly a big amount of transients (like Drums) and SF minima values remain above a certain value then the width will vary with the gain reduction a lot, while if the SF minima reach lower values then the width will vary a lot less with the gain reduction:

$$\log|\mathrm{width}|[\mathrm{n}] = \mathrm{m} \cdot cv_{\mathrm{Avg}}[\mathrm{n}]^{\mathrm{k}}$$

The **m** will be retained as a scale factor and we keep it at the same value. $\boldsymbol{k}$ is another scale factor carefully chosen to help the automation fit the results obtained from the evaluation that was conducted by Massberg as part of his project [2]. If $\boldsymbol{k}$ is 1 then we take the linear relationship that Massberg used, but if $\boldsymbol{k}$ takes smaller values then we get more interesting polynomial forms like the ones presented in figure 38. Some of these polynomials fit better on the evaluation data for the different instruments from [2]. Detailed presentations of the findings from the evaluation, as well as the interpretation of these findings that led us to defining the equation above are included in paragraph 4 of this paper.

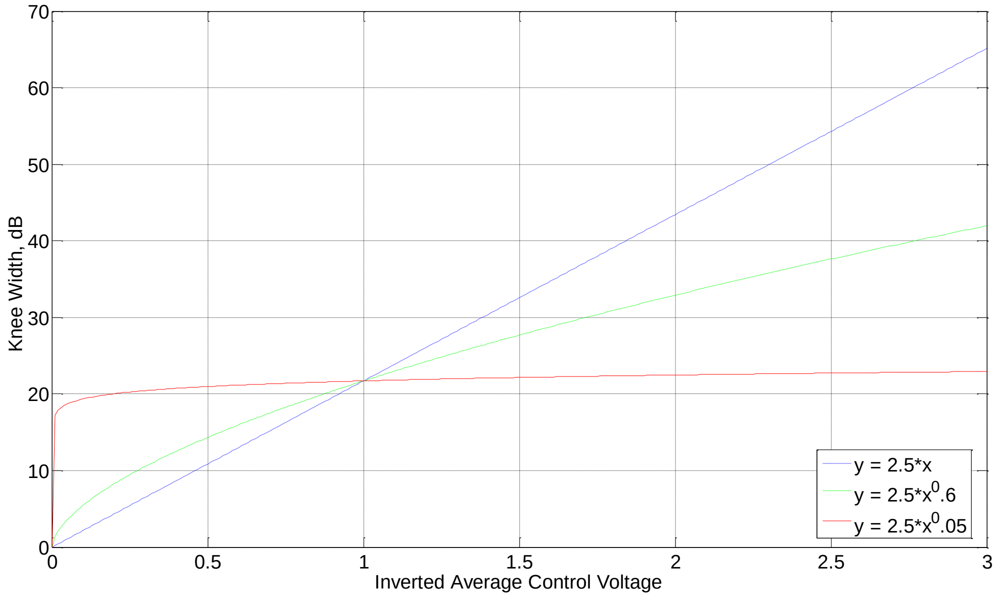


Figure 38. Knee Width over Average Gain Reduction functions. y corresponds to $\log|\text{width}|$ and x to cv_{Avg}.

As we will see in chapter 4, the polynomial, represented by the red line in figure 37, fits a lot better the evaluation data for the knee width for the Bass sample case, while the polynomial represented by the green line fits the Drums sample case. The blue line (linear case) that Massberg used does not fit well in any case so can probably hypothesize that to the human ear a linear increase of the knee width with the gain reduction (control voltage) does not sound well.

What remains is a method to be able and set the scale factor **k** to 0.05 if the sound is similar to the Bass sample or to 0.6 for a Drums-like sound. That side information about the signal can again arrive from the spectral flux computation. This time we will not focus on the minima of the spectral flux function for each signal and not on the peaks. The idea is that a signal that is composed mainly from transients, like a Drums sample, will never present very low values for minima since there will always be transient “activity” captured by the spectral flux. On the contrary, a signal like bass, even slap bass, will have the spectral flux minima reaching lower values since the initial transients of the attack part of the notes will quickly fade out while the steady-state part will remain longer. To illustrate our point we will present a series of figures (39-42) for various samples with the normalized spectral flux, the minima detected by inversing the function and applying an instantaneous attack peak detector and an average of the minima obtained by passing the peak detector output through a low-pass averaging filter to smooth out the changes. Here it is worth mentioning that initially we tried the idea of extracting the average of the spectral flux by passing it through a low-pass filter, but unfortunately, this method did not provide any interesting results that could help us identify between the various signals since almost all of them had similar spectral flux average values.

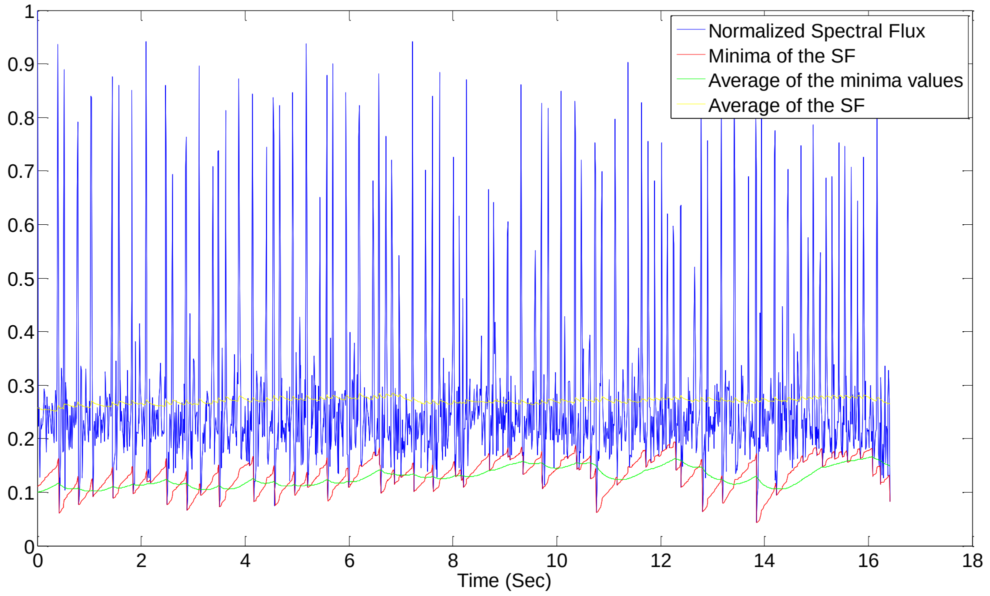


Figure 39. Spectral Flux of the Drums sample

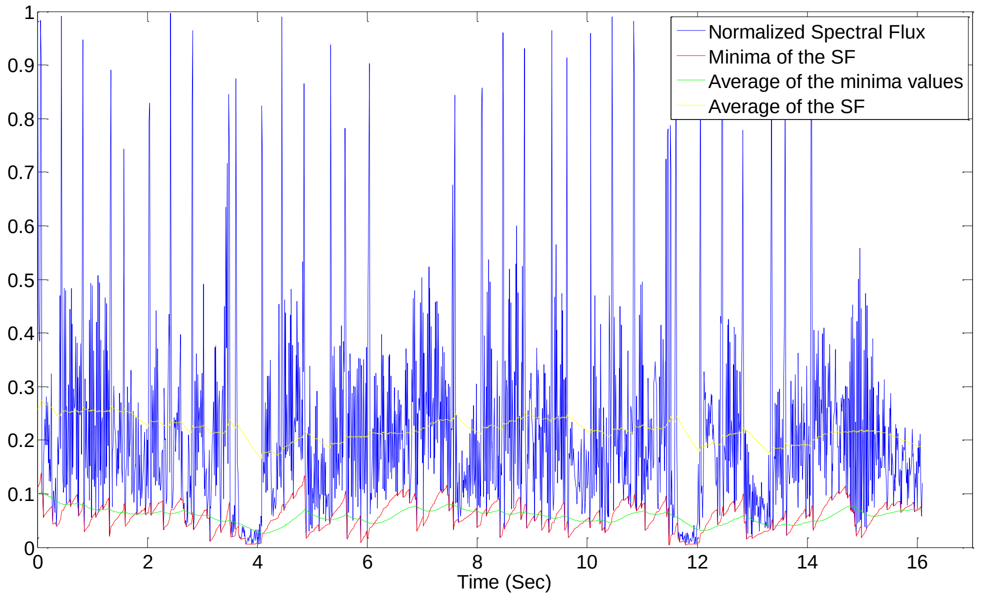


Figure 40. Spectral Flux of the Bass sample

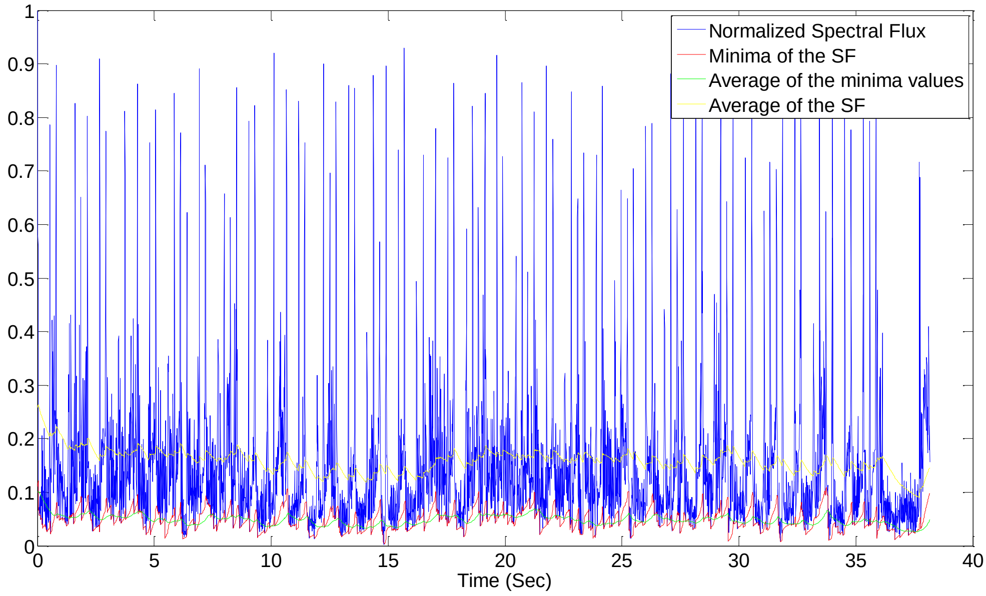


Figure 41. Spectral Flux of the Guitar sample

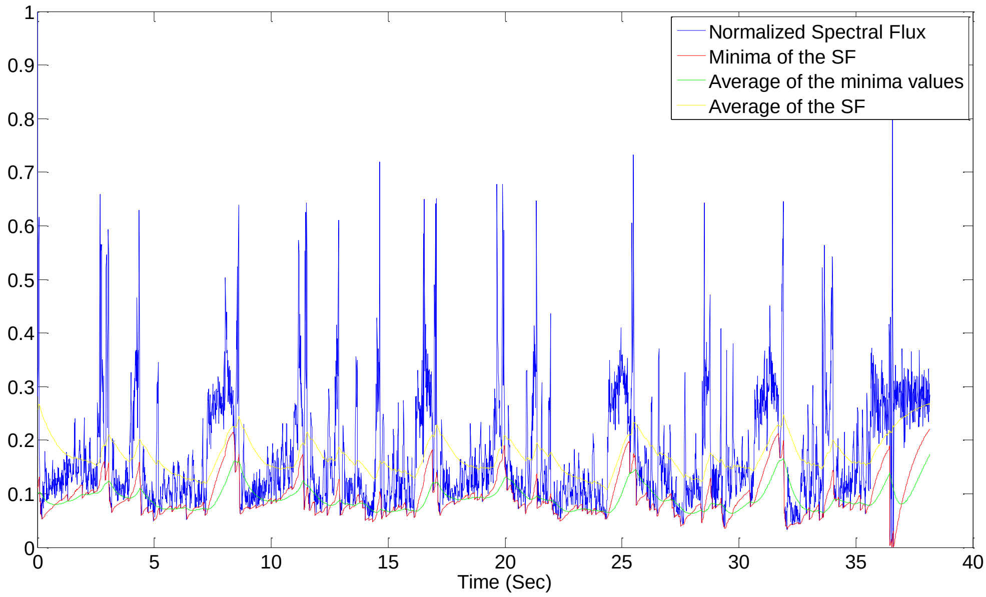


Figure 42. Spectral Flux of the Vocals sample

As it can be seen from the figures, Drums have the highest values for the SF minima, followed by the Vocals which also have a relative high amount of transients mainly because of the vocal tract and the way phonemes are shaped. Following the Vocals, is the Bass, which has a transient part because of the slap technique in addition to the steady-state part and in the last place is the guitar.

Now with a few lines of code we can simply correspond the information from the average of the spectral flux minima to a $\boldsymbol{k}$ factor value for the knee width equation as this was presented earlier. We simply set the $\boldsymbol{k}$ factor value to a set value if $\boldsymbol{SF\ minima_{Avg}}$ is above a threshold value or we change it to another value if the $\boldsymbol{SF\ minima_{Avg}}$ crosses the threshold, so:

$$\boldsymbol{k} = \begin{cases} \mathbf{0.6} & \boldsymbol{if\ SF\ minima_{Avg}} > 0.1 \\ \mathbf{0.05} & \boldsymbol{if\ SF\ minima_{Avg}} \leq \mathbf{0.1} \end{cases}$$

We could make the relationship more complex and add more $\boldsymbol{k}$ values since the average of the SF minima for the drums is always above 0.1 and for the most time even above 0.12 while for the bass it is constantly below 0.8. So we could make use of the in-between range but since we do not have evaluation results for any of the other signals we can just keep it simple for now. The method performs very well and as we will see in chapter 4 in a direct comparison with the human preferences it gives fitting results.

Finally, like with the auto attack and release case, if one wants to use the method in a real time implementation they will have to add a delay based on the window length of the spectral flux and also find a way like the ones we introduced in the attack and release times section to use the spectral flux value for all the corresponding samples inside each window.

CHAPTER 4: RESULTS AND DISCUSSION

As part of his paper [2], Massberg included an evaluation for all the methods we presented and/or designed. The evaluation was done in the form of a listening test between two groups of people. One consisted of 9 experts in the field of mixing, like mixing engineers (The Professionals group), the other group was consisted of 7 less experienced individuals, like students, audio researchers, hobby musicians (the Amateurs group).

The test was not performed using the same methods as most listening tests on the audio field are done. Instead of a subjective evaluation including comparative tests with pre-processed samples, in order to reach qualitative results concerning the auto compressor's performance, the test tried to collect quantitative data on how humans will set up and use a dynamic range compressor in their environment with their own equipment and whether the results agree with the automation. This was achieved by sending out a VST plug-in of the auto compressor with 4 short audio tracks of drums, bass played in "slap style", soft vocals and acoustic guitar and test instructions. The instructions included a series of listening tests in which the users had to tune individual parameters to their preferred setting while keeping all the other parameters fixed at specific settings, predefined in the instructions. The predefined values were usually such that would generate obvious amounts of compression and make any compression artefacts easily spotted by the listener. The findings of the test where later compared with what the automation method had chosen as preferred automated parameter setting.

4.1 Questions Arising from the Evaluation

The way the evaluation was performed, leaves room for a lot of debating and questioning on the methods used. A lack of restrains and subjectivity for the listening test will result in a wide-spread range of results (as it did) most of which do not tell us much about human preferences. Even Massberg himself characterised the nature of the test as "unorthodox" and addressed some possible issues and peculiarities that might come up from it [2]. The most important issues he addressed were the following:

- The number of testers (data set) is very small and possibly not representative.

- The number of test signals is very limited and even though it was quite diverse it is not representative of each instrument and its dynamic range of sounds it can produce.

- The test was performed on individual tracks that were not part of a general mix. As a result what was tested was not compression of tracks in order to nicely fit into a mix, but rather individual track compression which is very hard for someone to decide on what they actually want to achieve using compression. Do they want heavy compression, to soften a sound, add some punch or do some soft peak trimming for a subtle result? Individual track compression is a lot more susceptible to being affected by the artistic intention of its user.

- The tests were performed by the testers in a non controlled environment without supervising and using their own systems and equipment (speakers etc.). The validity of the results cannot be guaranteed since mistakes might have occurred. Furthermore, it is impossible for one to predict the influence the different listening environments and equipment had on each tester's choice of favoured settings. The influence might have been big enough that has caused the range of results to widen by a significant factor.

- Finally, the preferred human choices for each setting had to be compared against the compressor's automation method. But while the first ones are single, static values, the automation is an adaptive method, producing different values for each sample instead of a fixed value. Massberg performed a static comparison by

All these issues are to be considered when studying the results of the evaluation. Unfortunately, there was not enough time to perform a new better organized and thought of evaluation test. And apart from that, we wanted to use what meaningful conclusions were made from the evaluation in [2] to improve our automation methods. However, we have to argue that a subjective test, based on ranking various audio samples in order to extract qualitative results on preferred parameter settings and compare them with the automated method to see how well the automation is ranked, would have been a much more efficient method with more useful results.

Even if someone insisted in going for a similar evaluation method as that in [2], in order to collect quantitative data on how users prefer to set up a compressor, they should have restricted the evaluation method to be performed in a single specific listening environment and supervision. Trying to evaluate how well an automated compressor does compared to human preferences, one should safeguard that the automated method was judged ineffectual because the listening environment of the testers happen to affect the sound in a significant way to alter the results. Massberg wanted to obtain data on how people use a compressor, but applying no restrictions has as a result the data for each parameter being widespread and therefore, complicate the extraction of meaningful information.

Finally, one thing to consider is the method used to obtain a single representative value out of the vector of values from the automated method for each compression parameter.

4.2 Evaluation of the Auto Attack and Release Times

For the time constants evaluation test [2], the other parameters had to be predefined as follows: threshold at -30 dB, ratio at ∞:1 and knee width at 0 dB (hard knee). The two figures below present the test results, the first for the attack and the second for the release time, in a series of box plots. The box in each column indicates the interquartile range. The bottom of the box indicates the lower quartile (25th percentile) and the top of the box the upper quartile (75th percentile) while thick horizontal line within the box shows the median value of the data set.

The vertical black line passing underneath the box shows the sample range from the minimum to the maximum sample value. The thick black dot indicates the approximate value corresponding to the automation method.

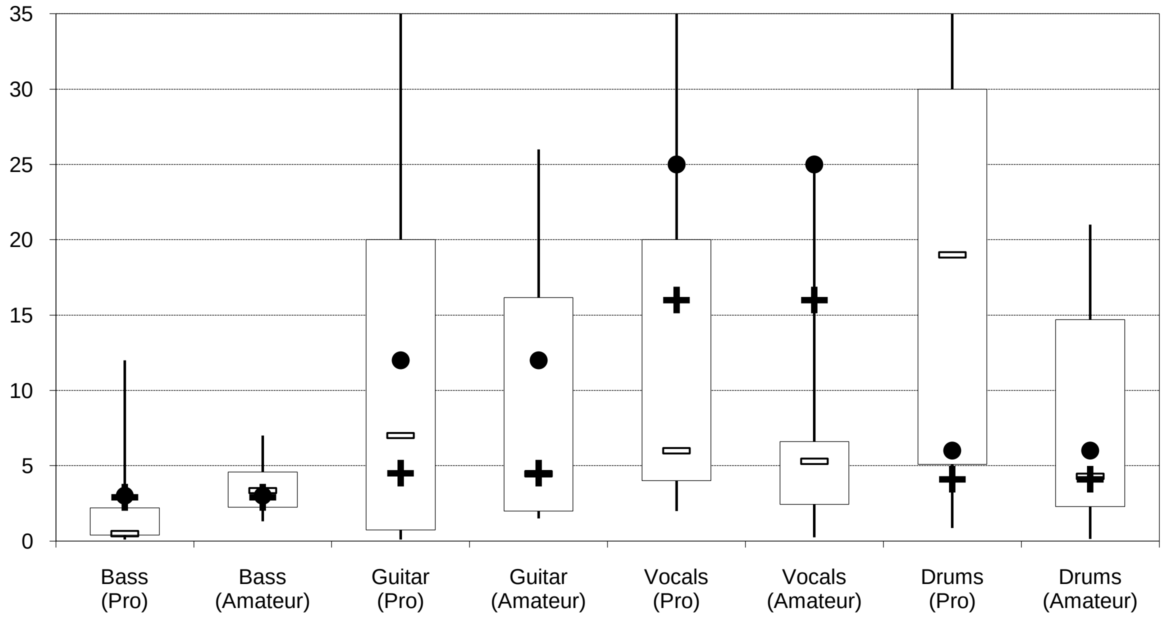


Figure 43. Box plots for the Evaluation test for the Attack Time [2]. Results in ms with median value (dash), Massberg's automation (dot) and our automation (cross)

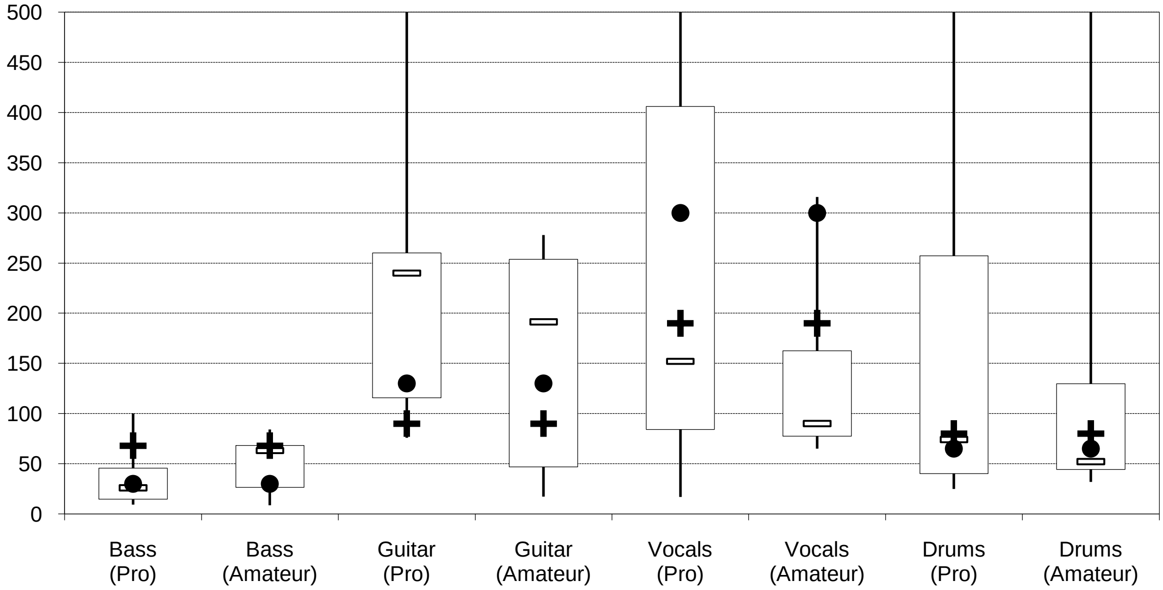


Figure 44. Box plots for the Evaluation test for the Release Time [2]. Results in ms with median value (dash), Massberg's automation (dot) and our automation (cross)

With the exception of the bass sample, for all other audio samples the choice of the testers was so diverse that the interquartile ranges (boxes) are quite big. This is strongly correlated to all the issues addressed in section (4.1). The analysis of the test results for the testers is presented in detail in [2], so instead of repeating it in this paper we will engage in presenting how our optimized methods fit in these figures.

We should remind here that Massberg used a very high value for the crest factor time constant. As a result, he got a very smooth crest factor with very small peaks at note onsets. At this point, we can argue that a less smoothed version of it, that would be able to catch transients faster and give higher peaks could work equally well, as this has already been tested in [31] with success. We can even claim that a more “peaky” function would give a better range of crest factor values and thus there would be no need to use the square of it when calculating the time constants. Furthermore, the use of some smoothing operation could smooth out “rough” parts in the function. Nevertheless, since the evaluation was conducted with a very smooth crest factor we also choose to follow the same approach and to smooth the spectral flux with the use of a peak detector, the attack and release time of which was fine-tuned to give desired results as explained in 3.3.7. If we had the luxury of more time, we could try a set of experiments-evaluations and find the optimal version of the spectral flux that can be used effectively in the calculation of attack and release times.

A problem we faced during the comparison of our results with that of Massberg was that it was impossible to choose a single value out of the various ones that the automation produced. Massberg in his project [2], used a method, he described as highly unscientific. In the method, he used a copy of the auto compressed signal in one track but with its phase inversed and compared it with a signal he manually compressed in another track with the same settings apart from the parameter of interest (attack or release), which he adjusted manually until he achieved a high phase cancellation. Since we cannot do the same, we decide to do a more straightforward and correct method. For the attack time we calculated the mean value out of all the attack time values that fall in a time period equal to the maximum attack time after every onset (peak of the spectral flux) of the signal. For the release time, since we cannot predict exactly at what point in time the release time values will be used we simply found the mean out of all the values. After all ,release times were a lot less varied than the corresponding attack ones because of the different release time constant we used for the peak detector.

Even now though, the method should be only roughly compared with that of Massberg's and the choice of the testers since all three were calculated in different ways. From the findings of the evaluation, looking at figures 43 and 44 we can say that our method performs a bit better than Massberg's but definitely there is space for improvement in the method. Overall, we managed to improve the results for the attack and release for the Vocals sample and for the attack times for the Guitar case. The same did not happen for the release times for the guitar since we actually got a bit worse results there. The results for the bass and the drums samples have remained at the same, already very good, levels. The professionals group asks for a bit longer attack times for the bass case but, especially after comparing this with the amateur choice, we can claim that this is mainly due to artistic intentions to make the drums sound more "punchy" and defined by leaving the very first transients of the notes to slip through the compressor.

For a more direct comparison we have included a set of figures (50 to 54), in the appendix A, with the time constants chosen by our automation method and that of massberg's. Massberg's method varies from smooth in some signals (i.e. Bass) to quite rough in others (i.e. Vocals, Guitar). Our method is far more stable and the results are closer to how a human operator would vary the parameters if they were given the possibility to do so.

4.3 Evaluation of the Auto Knee

The predefined parameters for this evaluation test, [2], were: the ratio at ∞:1, the attack time at 0.5 ms and the release time at 100 ms. The testers were then asked to choose their preferred knee width for three different threshold values: -18 dB, -25dB and -40 dB. The threshold can be seen as a function of the amount of compression applied on a signal, since the lower the threshold the more part of a signal will be compressed. For this evaluation test, Massberg concentrated on the drums sample and on the bass sample given that, both being more percussive, the influence of the knee is much easier to be heard on them. The results of this test are presented on figures 45 and 46. As before the thick dot is Massberg's automation method, the thick dash is the median of the human preference for each case, and the cross is our automation method.

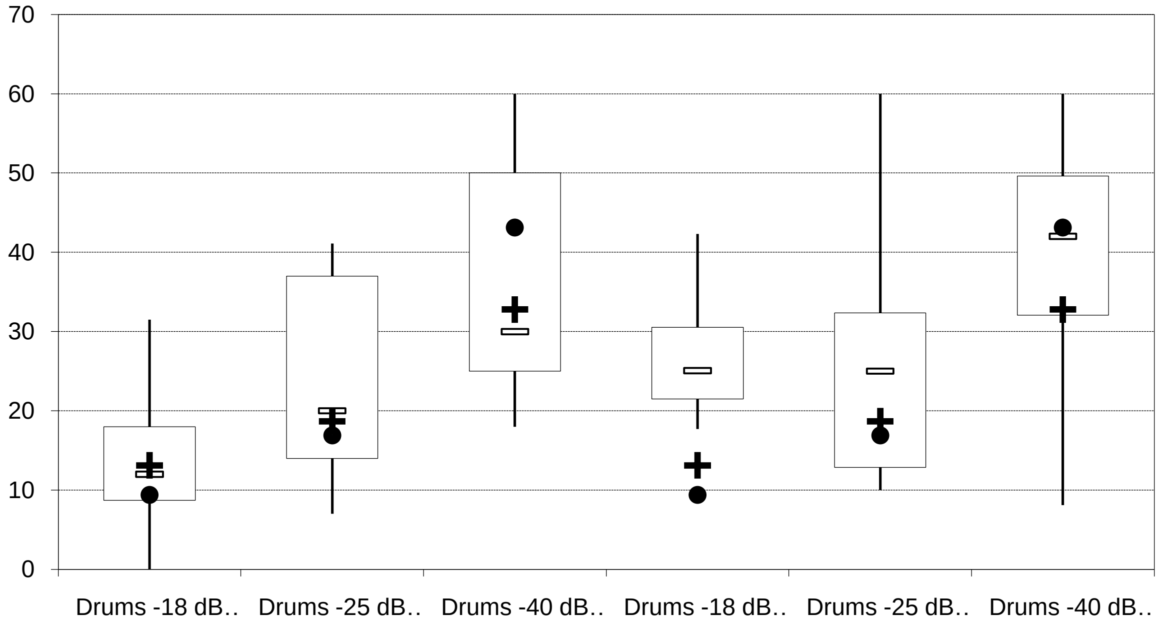


Figure 45. Box plots for the Evaluation test for the knee width. Drums sample – results in dB with median value (dash), Massberg's automation (dot) and our automation (cross)

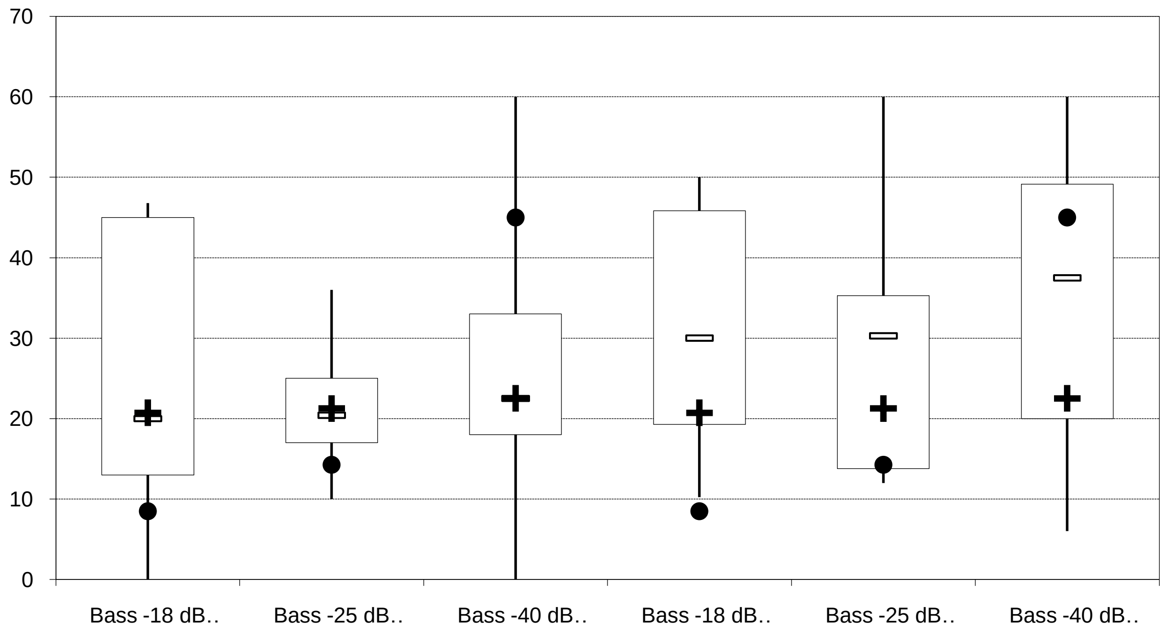


Figure 46. Box plots for the Evaluation test for the knee width. Bass sample – results in dB with median value (dash), Massberg's automation (dot) and our automation (cross)

After carefully examining the results, we can say that, the assumption Massberg made, that heavier compression can benefit from a softer knee [2], holds true for the Drums sample. Unfortunately, it is not valid for the slap bass sample, even though the signal is still very percussive. In the bass sample the median for the different threshold values remains fairly constant. This suggests that the testers preferred to retain the knee width at the same length even when the amount of compression on the audio sample was increased.

Massberg's method also fails at another point. Concentrating on the drums sample figure in the results for the professionals (since these are more credible) we can observe that the knee width increases to very high values as the compression amount increases (threshold goes down). As a result for a threshold of -40 dB the value of the automation is already a lot higher than the median. In Massberg's automation method, the relation between the average control voltage (directly relative to the amount of compression) and the knee width is linear with a slope of 2.5. But the aforementioned findings suggest that the slope is too high, and in conjunction with the bass sample results they suggest that the linear relationship is not correct but a higher order polynomial should be used to relate the knee width with the amount of compression.

On the other hand, the equation we proposed for our automation method for the auto knee was carefully chosen to fit on the data of the evaluation test. Apart from that it was also tested in other signals to ensure that it performs satisfactory with them and avoid the danger of overfitting the data of the evaluation in [2]. Since we do not have any evaluation data for other signals we cannot check how well it performs in general but we did safeguard that it gives reasonable and partly expected results. A more thorough evaluation test with more signals will give more insight on the method and minor alterations might be introduced to it, to make it work more effectively over a wider range of signals.

4.4 Evaluation of the Auto Make-up Gain

For the Make-up gain evaluation test [2] the predefined parameters were chosen to be: threshold at -30 dB, ratio at ∞:1, knee width at 0 dB (hard knee), attack time at 0.5 ms and release time at 100 ms. These settings (low threshold and especially the very short time constants) were chosen as such to guarantee that all 4 test samples will be heavily compressed with not even short transients managing to slip through and their dynamic range will be greatly reduced after compression. The testers had to manually vary the make-up gain control to a level they were satisfied with, trying to make the compressed signal being equally loud to the uncompressed. The results can be seen in figure 47, below.

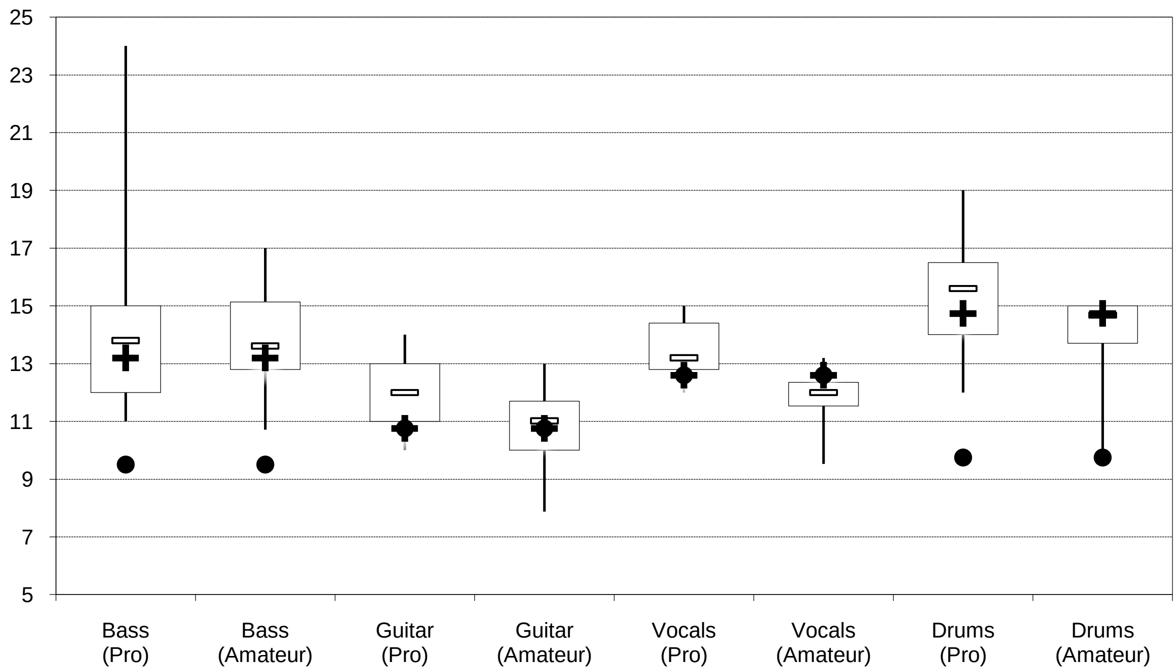


Figure 47. Box plots for the Evaluation test for Make-up gain. Results in dB with median value (dash) and Massberg's automation (dot).

Even in this experiment, there is still a significant spread among the choices of the testers and in some cases the results of the amateurs suggest a lower make-up gain by 1 or 2 dB. This can partly be due to the unsupervised and uncontrolled listening environment conditions but as Massberg mentions in [2] it is more probably due to the big difference between the dynamic ranges of the compressed and uncompressed signal that made the process of making them equally loud quite hard.

Nevertheless, the interquartile range in each case is no bigger than 3 dBs and examining Massberg's method we can see that it performs well for Guitar and Vocals but it differs a lot for the Drums and the Bass samples. Furthermore, in all cases apart from one, his automation suggests a make-up gain value lower than what the testers were satisfied with. It seems that when a signal's dynamic range is reduced, the signal is perceived as less loud than before even if we apply on it a make-up gain equal to the average of the gain reduction of the signal from the compressor.

Comparing the loudness of the signal before and after compression using equal loudness filters was not sufficient to explain why people would ask for more dB of make-up gain than the actual average gain reduction in order to reach the same perceived loudness. The loudness models increased the make-up gain by 1 db at most while in samples like the bass and the drums, we need at least 4 dB to match the median of the testers. On the other hand measuring the time duration of the compressed peaks for each signal we found out that the average time durations for the bass and the drums samples were far smaller than that of the guitar and the bass signal and also below the value of 200 ms. Figure 48 presents a table with the average values for a compression case with the same parameter settings as used for the evaluation test of the make-up gain. Based on that, we came up with a simple method to give these two signals a boost in the make-up gain (see section 3.5.5). And indeed after this boost their make-up gains are improved greatly and are close to the human preferences. The only drawback of the method is that it can only work online as it is based on finding the average peak duration of signals, but it might be possible to be improved to work over a short time windows and calculate time duration of single signal peaks.

Audio Samples	Peak Average Time Duration (ms)
Bass	56
Drums	97
Guitar	319
Vocals	546

Figure 48. Average Time Duration of Compressed Peaks in Different Audio Samples

CHAPTER 5: CONCLUSION AND FURTHER WORK

In this project report, we initially presented and explained the audio effect of dynamic range compression. We analyzed its various stages and examined how they affect the overall performance of the effect. Afterwards, we proposed a compressor design based on suggestions from the academia and various references after incorporating our own modifications. In the desing process we included automation methods to automate most of the compressor parameters, proposed by M. Massberg in his paper [2]. We studied the performance of these methods and finally, we investigated alternative methods that would improve and expand the automation process. We compared our proposed methods against the suggestions from [2] as well as against the choices of human operators that were available from the evaluation test in [2].

The idea to use the spectral flux as opposed to the crest factor to automate the attack and release times was considered successful. Spectral flux is much more sensitive than the crest factor in capturing the signal’s onsets and transient content and thus it can give more accurate results. Nevertheless, even with the spectral flux it is impossible to automate creative tasks related to the artistic intention of the user, and both the attack and the release time are the parameters through which the user will introduce specific artefacts to shape the sound of the signal based on his intensions.

The modification we introduced in the auto knee method, to include information from the spectral flux in the auto knee width calculation, lead to a successful improvement with very good results. We managed to follow the choices of the human operators very closely. It is still necessary though to perform more evaluation tests with a bigger range of audio signal samples.

The introduction of a loudness measure for the auto make-up gain was not crowned with success as it did not give the expected results. The increase in the make-up gain that would have brought the automated result closer to the human choices was insignificant. On the other hand, the time-based approach gave interesting results and improved the make-up gain in the bass and the drums samples, where it was most problematic.

Furthermore, we suggested a different approach to the automation of the static compression characteristic. Instead of setting the ratio to ∞:1, simulating various ratios with the varied knee width and leaving then threshold to be manually adjusted by the user, an alternative would be to automate the threshold to follow the RMS of the signal and let the user adjust the threshold based on what they prefer. Doing this, we avoid keeping the ratio fixed at infinity which according to Massberg in [2] makes the compressor sound aggressive and ‘limiter-ish’ even with the choice of a very soft knee. The method we proposed was not directly tested due to inadequate time, but it has been successfully used in [31].

5.1 Further Work

We would like to believe that this report is a step forward for both the field of dynamic range compression and the field of automating digital audio effects. There is only a small amount of papers in academia that concentrate in compression so we want to believe that this report will help expand the work on the field.

As far as this report goes, the attempt was quite challenging but we believe that we came up with promising results given the very limited time at hand. Further work can be put in the automation methods for the compressor and some of the suggested ones can see further development and optimization.

Further investigation is needed for the case of the make-up gain. The secret to achieve better results, closer to the human preference is probably hiding still in loudness and the perception of sound by the human ear but it should be approached in another way than the one we used. The idea we proposed that takes into account the length of the peaks in a signal gave very good results. So possibly, further work and optimization on that method, together with weighting louder parts of the signal might prove to be the best solution.

Spectral flux and the way we use it to automate the attack and release time can also be investigated further and reach an even more optimal result. A further improvement for the method might be to include the amount of gain reduction in the calculation of the attack and release time since a heavily compressed signal with huge amounts of gain reduction might require longer time constants. Also, the idea of using machine learning techniques to train the compressor in discriminating between instruments so as to be able to recognise the input signal and adjust its automation to it might also be a great improvement although it would be quite challenging to design.

Finally, a more thorough evaluation under supervised conditions should be conducted to obtain a bigger and more reliable amount of data on human preferences for the compression parameters. This would help to fine-tune all the automation methods and ensure they behave well over a wide-range of signals, while they do not overfit on the data.

REFERENCES

[1] Izhaki, Roey. (2008). *Mixing audio, concepts, practices and tools.* Focal Press. (ISBN 9780240520681)

[2] Massberg, M. (2009). *Investigation in Dynamic Range Compression*. MSc project report at Queen Mary University of London.

[3] CCRMA, Stanford. Dynamic Range Compression. Retrieved from the Stanford website: https://ccrma.stanfrod.edu/~jos/fp/Nonlinear_Filter_Example_Dynamic.html

[4] Queen Mary University, Compression and Limiting Lecture on Digital Audio Effect Module, presented by Dr. J. D. Reiss. 2009-10

[5] Wikipedia. Loudness War. Retrieved from Wikipedia web site: http://en.wikipedia.org/wiki/Loudness_war

[6] Tyler, L.B. (1979). *An Above Threshold Compressor with One Control.* Presented at the 63rd AES Convention 15-18 May, 1979. Los Angeles.

[7] Abel, J.S. & Berners, D.P. (2003). *On Peak-Detecting and RMS Feedback and Feedforward Compressors*. Audio Engineering Society, Convention Paper 5914. Presented at the 115th AES Convention 10-13 August, 2003. New York.

[8] Simmer, U. Schmidt, D and Bitzer, J. (2006). *Parameter Estimation of Dynamic Range Compressors: Models, Procedures and Test Signals.* Audio Engineering Society, Convention Paper 6849. Presented at the 120th AES Convention 20-23 May, 2006. Paris, France.

[9] Presented by Glenister N. (2004) *Frequency Weighting Equations*. Retrieved from Web site: http://www.cross-spectrum.com/audio/weighting.html

[10] Bello, J. Daudet, L. Abdullah, S. Duxbury, C. Davies, M. and Sandler, M. (2005). *A tutorial on Onset Detection in Music Signals*, IEEE Transactions on Speech and Audio Processing, vol. 13, no. 5, pp. 1035-1047, September 2005.

[11] Dixon, S. (2006). *Onset Detection Revisited*, Proc. Of the 9th Int. Conference on Digital Audio Effects (DAFx-06), Montreal, Canada, September 18-29, 2006.

[12] Klapuri, A. (1999) *Sound Onset Detection by Applying Psychoacoustic Knowledge*, in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, Phoenix, Arizona, 1999.

[13] Zaunschirm, M. Reiss, J. And Klapuri, A. (2010) *Dynamic Range Alteration Based on Transient Modification*, To be submitted to the Computer Music Journal.

[14] Nam, J. (2009) *The Chorus Effect – Simulation of Multiple Sound Sources Playing in Unison*, Project done as part of Music 220C module at Stanford University, CCRMA, Department of Music, Spring 2009.

[15] American National Standards Institute, "American national psychoacoustical terminology" S3.20, 1973, American Standards Association.

[16] Everest, F. Alton (1989) *The Master handbook of Acoustics*. 2nd ed. TAB Books (a division of McGraw Hill) printed in the United States of America. (ISBN 0-8306-9396-3)

[17] Robinson, D.W. and Dadson, R.S. (1956) *A Re-Determination of the Equal-Loudness Relations for Pure Tones*. British Journal of Applied Physics, Vol. 7, 1956, pp. 166-181. (Adopted by the International Standards Organization as recommendation R-226)

[18] Soulodre, G.A. Norcross, S.G. (2003) *Objective Measures of Loudness*. Audio Engineering Society, Convention Paper 5896. Presented at the 115th AES Convention 10-13 October, 2003. New York.

[19] Aarts, R.M. (1992) *A Comparison of Some Loudness Measures for Loudspeaker Listening Tests.* J. Audio Eng. Soc., Vol. 40, No. 3, pp. 142-146, March 1992.

[20] ISO 532-1975(E), *Acoustics – Method for Calculating Loudness Level*. 1st ed. 1975, International Organization for Standardization, Geneva, Switzerland (1977)

[21] Moore, B. C. J. Glasberg, B. and Baer, T. (1997) *A Model for the Prediction of Thresholds, Loudness, and Partial Loudness*, J. Audio Eng. Soc., Vol. 45, No. 4, pp. 224-240, 1997.

[22] Glasberg, B. R. and Moore, B. C. J. (2002) *A Model of Loudness Applicable to Time-Varying Sounds*, JAES Volume 50 Issue 5 pp. 331-342; May 2002

[23] Benjamin, E. (2002) *Comparison of Objective Measures of Loudness Using Audio Program Material*. Audio Engineering Society, Convention Paper 5703. Presented at the 113th AES Convention 5-8 October, 2002. Los Angeles, CA, USA.

[24] Seefeldt, A. Crockett, B. And Smithers, M. (2004) *A new Objective Measure of Perceived Loudness*. Audio Engineering Society, Convention Paper 6236. Presented at the 117th AES Convention 28-31 October, 2004. San Francisco, CA, USA.

[25] Grimm, E. Skovenborg, E. And Spikofski, G (2010) *Determining an Optimal Gated Loudness Measurement for TV Sound Normalization*. Audio Engineering Society, Convention Paper 8154. Presented at the 128th AES Convention, 22-25 May, 2010. London, UK.

[26] Cassidy, R.J. (2004) *Dynamic Range Compression of Audio Signals with Recent Time-varying Loudness Models*. Center for Computer Research in Music and Acoustics (CCRMA) Stanford University, Stanford, CA 9430

[27] Bemrsoe, R. (2007). *Louder Sounds Better.* Retrieved from Web site: http://blogs.msdn.com/b/audiofo,ol/archive/2007/02/07/louder-sounds-better.aspx

[28] Gonzalez, E.P. Reiss, J. (2009) *Automatic equalization of multi-channel audio using cross-adaptive methods*. Audio Engineering Society, Convention Paper 8154. Presented at the 127th AES Convention, 9-12 October, 2009. New York, USA.

[29] Timoney, J. Lysaght, T. And Schoenwiesner, M. (2004) *Implementing Loudness Models in Matlab*. Proc. of the 7th Int. Conference on Digital Audio Effects (DAFx'04), Naples, Italy, October 5-8, 2004.

[30] Discussion about '*Calculating sound amplitude in dB from a WAV file*'. Retrieved from DSPrelated.com Website: http://www.dsprelated.com/showmessage/29246/1.php

[31] Giannoulis, D. (2010) *Automated Noise Gate*. 3rd and Final Assignment for the module Digital Audio Effects, Lecturer: J. D. Reiss. Taught module for the MSc in Music Signal Processing, Queen Mary University, London.

[32] Z□ölzer, U. (2002) *DAFX Digital Audio Effects*. 2nd ed. John Wiley & Sons LTD. Printed and bound in Great Britain by Biddles Ltd, Guildford and King's Lynn. (ISBN 0-471-49078-4)

APPENDIX A

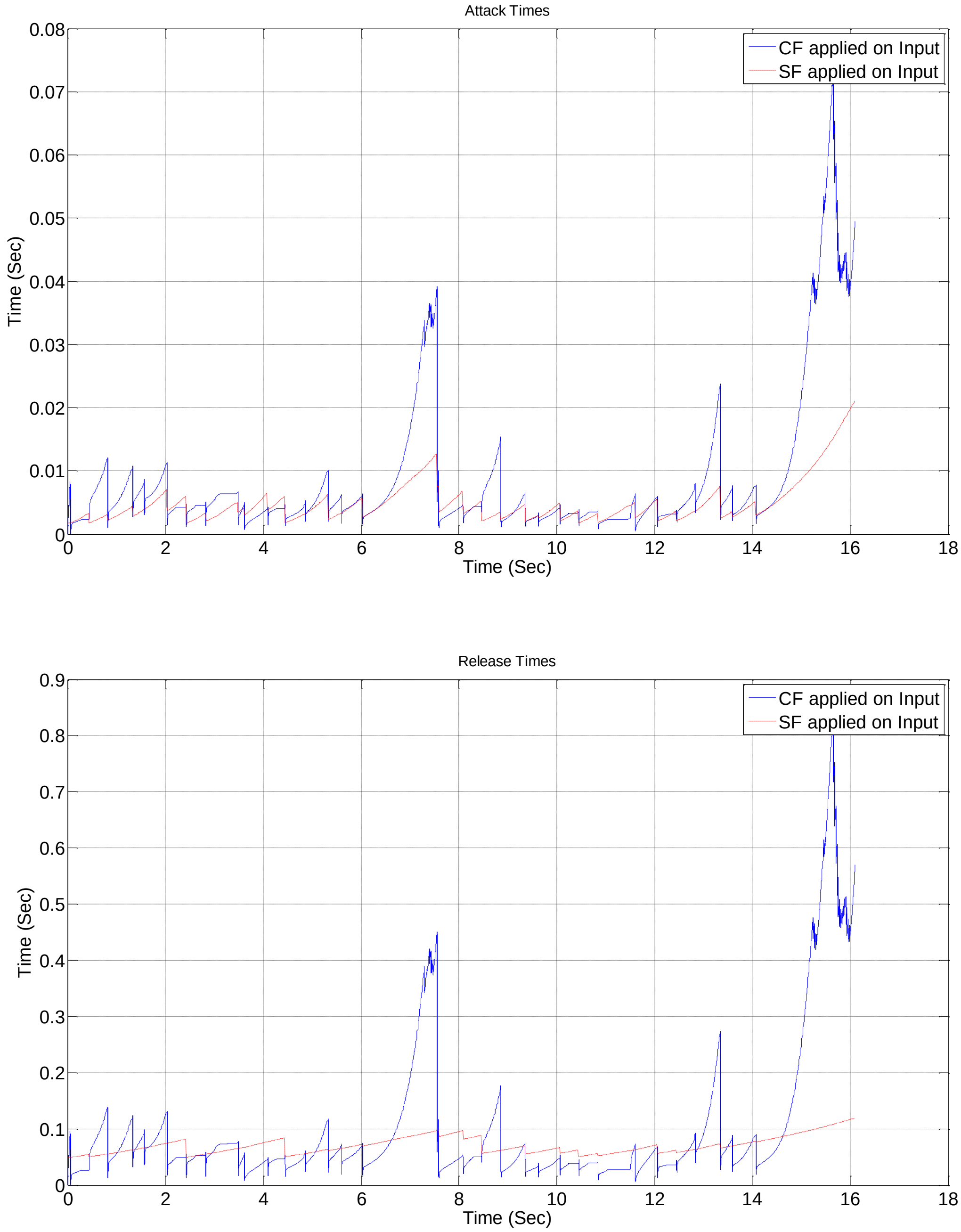


Figure 49. Attack and Release times for the Bass sample as this were calculated with the Spectral flux and the Crest factor method

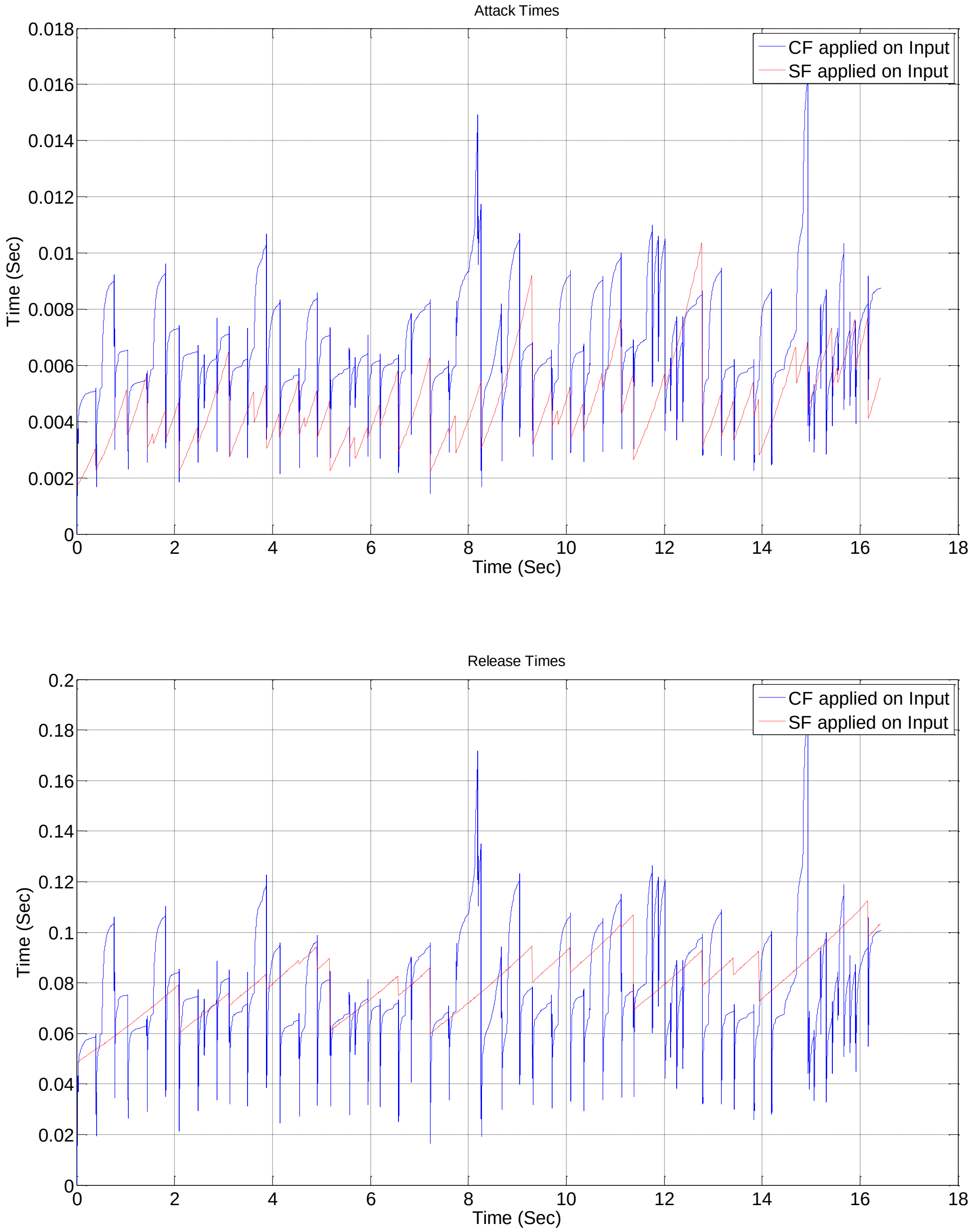


Figure 50. Attack and Release times for the Drums sample as this were calculated with the Spectral flux and the Crest factor method

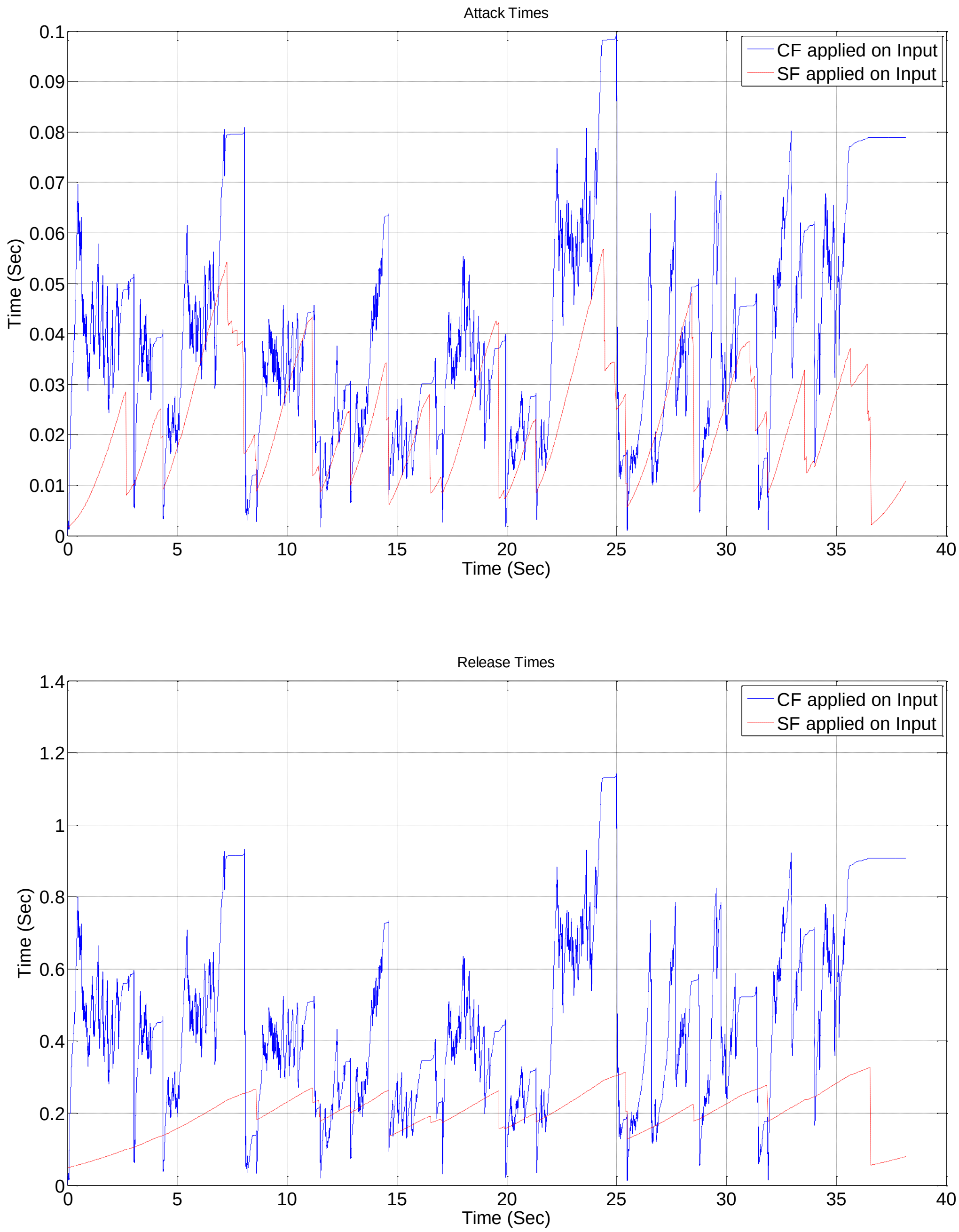
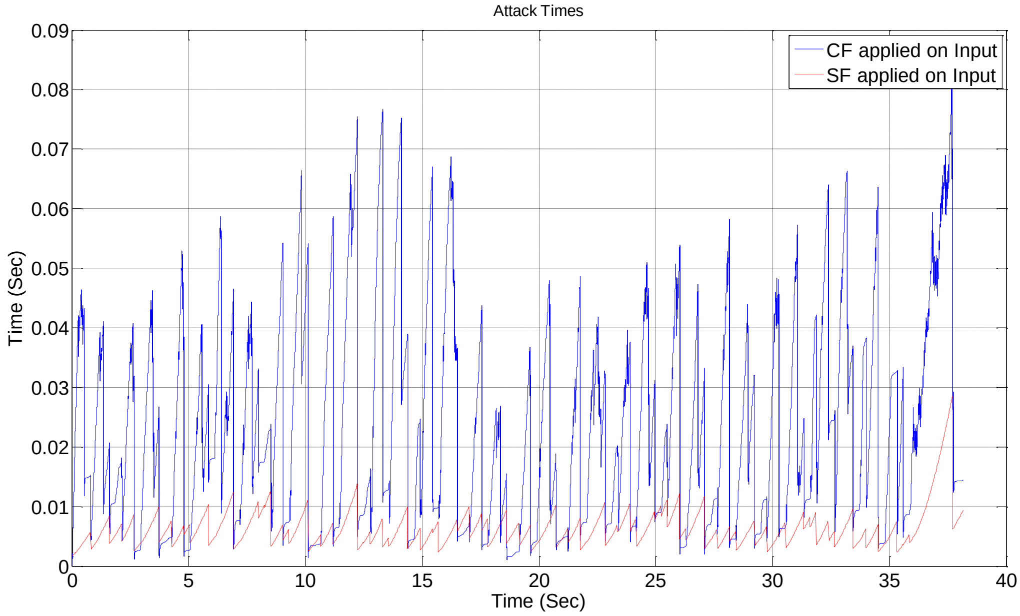


Figure 51. Attack and Release times for the Vocals sample as this were calculated with the Spectral flux and the Crest factor method



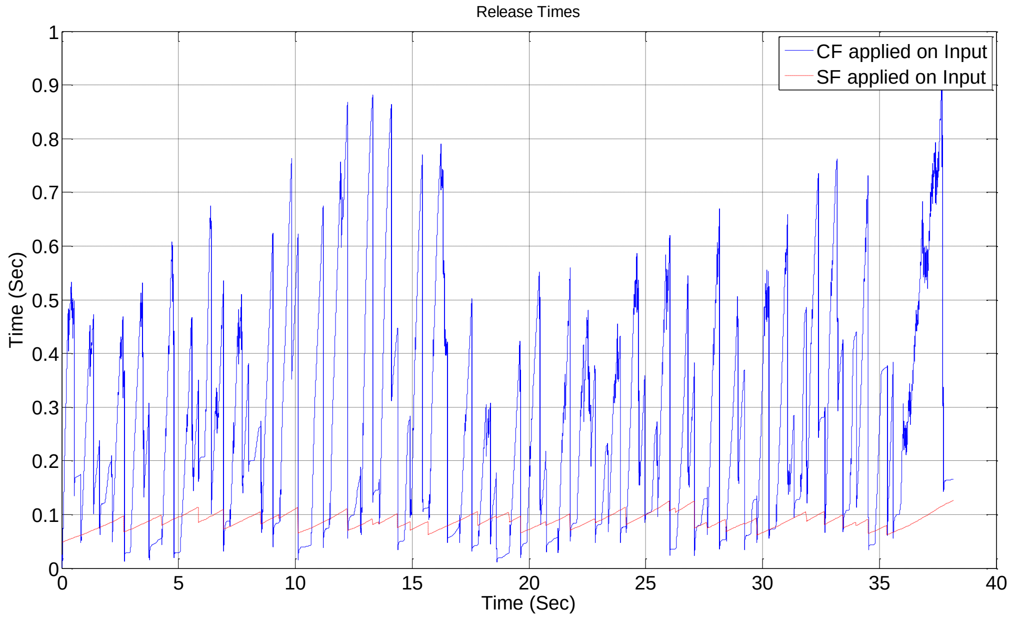


Figure 52. Attack and Release times for the Guitar sample as this were calculated with the Spectral flux and the Crest factor method