Audio Engineering Society
Convention Paper
Presented at the 121st Convention
2006 October 5–8 San Francisco, CA, USA

A Real-Time Rhythmic Analyzer and Equalizer

Jörn Loviscach[1]

[1] *Hochschule Bremen (University of Applied Sciences), Bremen, 28199, Germany*

Correspondence should be addressed to Jörn Loviscach (`jlovisca@informatik.hs-bremen.de`)

ABSTRACT

The rhythmic analyzer and equalizer presented in this paper allows to cut or boost the signal at a given audio frequency and a given rhythmic frequency, that is, number of beats per minute (BPM). A task that can be addressed with the rhythmic equalizer is for instance to emphasize series of 1/8 triplet notes played on the hi-hat of the drum set. The software works in real time and offers an interactive graphical user interface that supports both analysis and adjustment. The current energy distribution in the two-dimensional audio frequency (Hz) / rhythmic frequency (BPM) space is displayed as a continuously updated backdrop image. The user paints the intended adjustments of BPM levels and phases onto an image layer added on top of this image.

1. INTRODUCTION

Modulation spectra of low-level acoustic features have been used successfully in fields such as modeling audio perception [1], feature extraction [2, 3], and speaker recognition [4]. The rhythmic analyzer and equalizer introduced here applies the expressive power of modulation spectra to music production. In particular, it allows the user to not only view modulation spectra in real time and musically meaningful units, but also to tweak them, using a graphical interface for control. This user interface displays the analysis result using a xy diagram where every position corresponds to a certain rhythmic (i. e., modulation) frequency band (x) and audio frequency band (y), see Figure 1.

The overall architecture (see Figure 2) comprises

- an audio path composed of two channels times 40 bandpass filters and amplitude modulators acting in the audio frequency domain,
- 40 control processors, each composed of 120 bandpass filters, envelope detectors and other units acting in the rhythmic frequency range,
- a user interface for display and control employing current graphics hardware.

This paper is structured as follows: Section 2 describes the processing done in the audio frequency

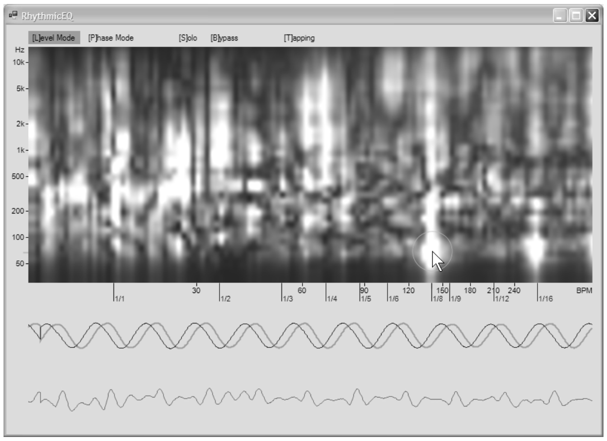

Fig. 1: The graphical user interface uses logarithmic axes, where x gives the BPM number and y the audio frequency in Hz. Per-band level details are presented as curves in the lower part.

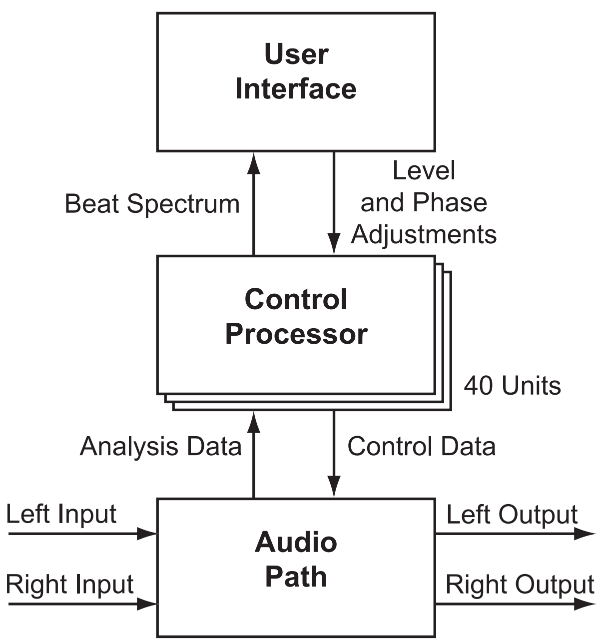

Fig. 2: The system is composed of three tiers.

range, Section 3 details the processing in the BPM frequency range, which includes the generation of control signals. How the analysis results are displayed and the user's input is used to control the equalization is covered in Section 4. Section 5 presents results; Section 6 concludes and points out directions for future work.

2. AUDIO PATH

Similar to what happens inside a standard narrow-band graphic equalizer, the audio range of both the left and the right audio signal is divided into 40 filter bands by banks of second-order bandpass filters with logarithmically spaced center frequencies ranging from 30 Hz to 15 kHz, see Figure 3. The width of the filters is chosen near one-third octave, which allows strong overlap and a flat frequency response when the output signals of all filters are added.

The splitting into bands is useful for a clean rhythm extraction; furthermore it diminishes pumping artifacts that may occur through the time-varying control to be applied. The number of 40 bands turned out as an attractive compromise between resolution on the one hand and computational load on the other. Furthermore, if the bands are chosen too narrow, a scale of notes may trigger different bands, but none of them rhythmically.

The combination of the audio filter bank with modulators and an adding stage acts similar to a multi-band compressor or a multi-band noise reduction scheme. To allow working in look-ahead mode, optional delay lines are inserted after each bandpass filter. Since rhythmic oscillations can be strongly boosted, a soft clipping stage is included in front of the outputs.

3. CONTROL PROCESSOR

Every set of stereo bands in the audio path is equipped with a control processor. It evaluates the audio signal level and generates a modulation signal taking the user's input into account, see Figure 4.

3.1. Analysis

The initial stage of each control processor consists of a stereo RMS detector with a time constant corresponding to 1/3 of the audio band's center frequency.

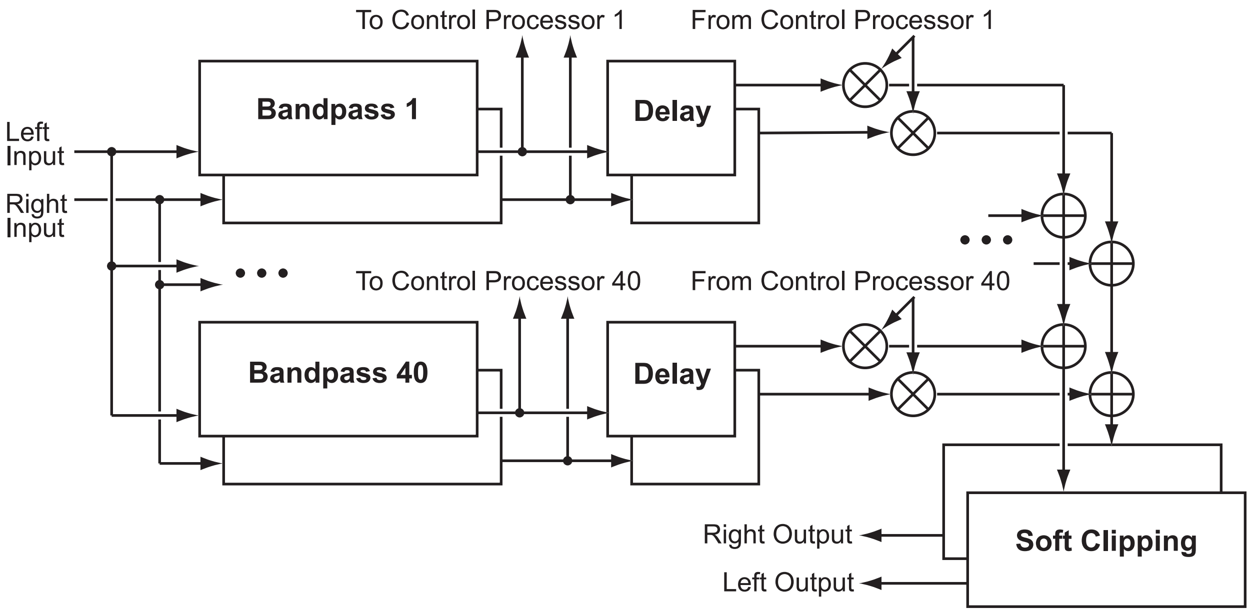

Fig. 3: Both the left and the right channel of the audio signal are split into 40 frequency bands.

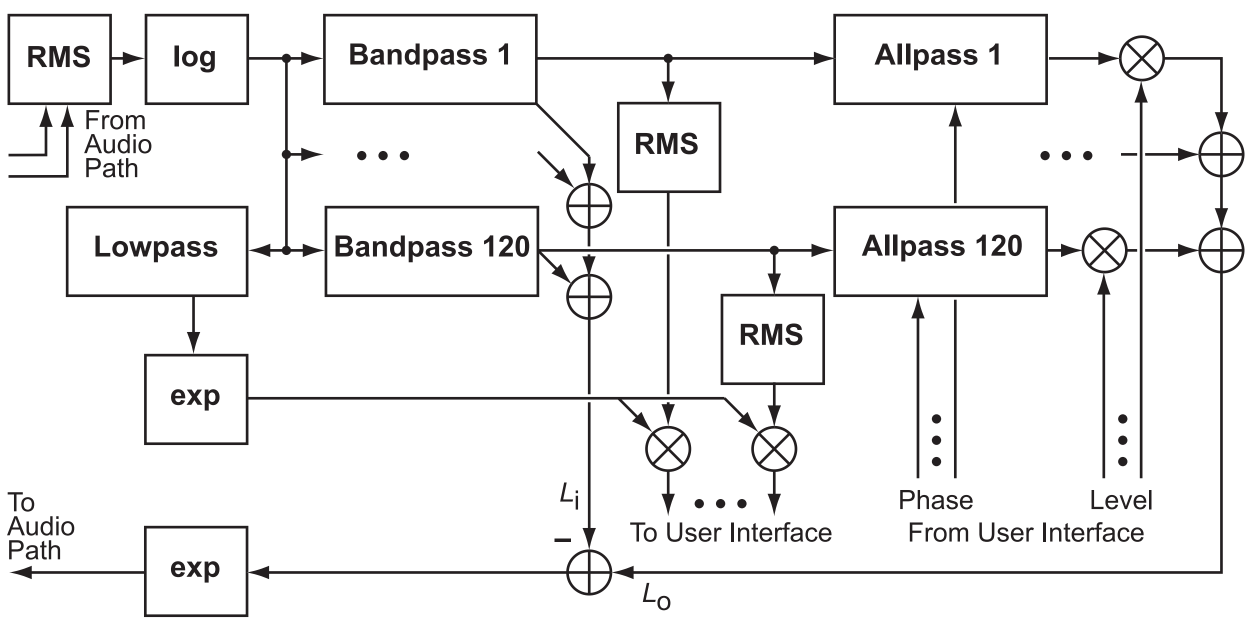

Fig. 4: Each of the 40 copies of the control processor contains 120 bandpass filters that act in the rhythmic frequency domain.

The mean-square signals are subsampled to 200 Hz. To lessen the computational load, all further computations in the control signal domain are executed at this reduced rate.

To convert the RMS amplitudes to levels, the logarithm is formed. The resulting level signal is subjected to a frequency analysis in the rhythmic domain. This analysis is accomplished through a filter bank of 120 second-order bandpass filters with center frequencies ranging from 0.17 to 6.7 Hz, or, equivalently, 10 to 400 BPM. Since the human perception of rhythm is mostly based on frequency ratios, the center frequencies are spaced logarithmically, similar to the audio domain. The large range and resolution of the filter bank captures rhythms extending over two bars in a medium tempo title as well as 1/8 notes in an up-tempo song.

3.2. Adjustment

The outputs of the rhythmic bandpass filters are used for both display and adjustment. For display, they are fed into RMS amplitude detectors with a time constant of 250 ms or a time constant corresponding to 1/2 of the rhythmic band's center BPM frequency, whichever is larger. This dependency on the BPM number extracts more—but not too much—temporal detail for faster beats.

The resulting 120×40 amplitude signals are fed into the display module. Before that, each amplitude signal is multiplied by the exponentiated average level of its spectral band, so that silent bands appear less emphasized in the display, even though they may contain as strong rhythmic oscillations as louder bands do. For adjustment, the outputs of the rhythmic bandpass filters are boosted or cut in their amplitude and then summed. Their sum L_o describes the level that the corresponding audio band should possess after equalization. To achieve this correction, the audio signal of this band has to multiplied by an appropriate, time-varying control signal c. Let L_i denote the initial level of this audio band as fed into the rhythmic analyzer. Then one can set

$$c = \exp(L_o - L_i).$$

If L_o is unchanged from L_i, the control signal c will be 1 and thus have no effect. If L_o is zero because all rhythmic bands have been cut completely by the user, c will act so as to produce a perfect compressor. To suppress any frequency or phase ripple that may occur due to the bandpass analysis stage, L_i is formed from the sum of the outputs of the bandpass filters, not from the level signal, see Figure 4.

Standard graphic equalizers for the audio domain does not offer phase control, in part due to the small effect that phase has in hearing. In the rhythmic domain, however, phase characterizes offbeat patterns and—more subtly—distinguishes between laid-back and driving grooves. Thus, the rhythmic equalizer is also equipped with a phase control for each of its 120×40 bands: Each output signal of a rhythmic bandpass filter is processed by a first-order allpass filter which shifts the phase according to the user's input.

4. USER INTERFACE

The visual interface employs current graphics hardware to display both the data and the user's adjustments. To decouple audio and graphics processing, the user interface runs in its own lower-priority processing thread, separated from the audio engine.

4.1. Display

The mass of data is easily displayed through a two-dimensional diagram, where the one axis denotes the BPM number, the other axis denotes the audio frequency, and every point (x, y) is colored according to the corresponding level. In the presented system, the resolution of the BPM numbers is higher than that of the audio range. Since user-interface windows typically are wider than taller, it makes more sense to use the x axis for the BPM numbers.

The 120×40 levels computed by the audio engine are converted to a grayscale image. To display as much as information as possible, the mean and the standard deviation of the levels are used to scale the conversion: The color range of the grayscale image comprises the mean plus/minus two standard deviations. One may think of employing a rainbow-like color scheme to clearly depict an even larger dynamic range. However, the range of the grayscale image turned out to be sufficient for standard tasks; on top of that, the system proposed here already uses coloring to display the adjustments to be made.

The image of 120×40 pixels is sent as a texture to the graphics card, as used in graphics programming to apply color patterns to 3D objects. A shader program running on the graphics card scales the texture to the window's full size, which is controllable by the user. In order to create a visually smooth image in this process, bicubic interpolation is applied. The central processing unit is not involved in these computations; a full-screen display can be refreshed approximately ten times a second without risking glitches in the audio processing. Higher update rates are not necessary, since the displayed data are effectively lowpass-filtered through the RMS level detector.

4.2. Input

The software prototype offers two mouse-paint modes: one to adjust the levels, another to adjust the phase shifts. The user marks boosts/cuts of the levels or positive/negative shifts of the phases with the colors green and red, respectively, in the xy plane. These colors are applied on top of the grayscale analysis image and processed by the same pixel shader on the graphics card. The paint tool is soft-edged; its radius—displayed as a circle—can be changed interactively with the mouse wheel; applied colors can be unpainted quickly.

For better control, the user interface also displays the control signals of the audio and rhythm band above which the mouse currently hovers in the xy diagram, see Figure 5. The upper two waves are the output of the rhythmic bandpass filter (gray) and the signal formed from it through phase shift and amplification/attenuation (black). The lower two waves depict the total level L_i of the audio band before equalization (gray) and its level L_o to be attained (black).

A Bypass switch lets the user compare the equalizer's output to the original signal. A Solo switch boosts the audio and rhythm band over which the mouse currently hovers and cuts all remaining ones. Switching off Solo returns the system to the level and phase settings that were in effect before.

The current solution does not aim at determining the music's tempo, that is: its BPM number in the listener's ear. The user can, however, easily set this tempo by tapping on the keyboard. With help of these data, markers are placed along the BPM axis showing where double measures, measures, half measures, quarter notes and so on are located. This allows for easy orientation. The user can also drag with the mouse to adjust the markers' placement. Due to the logarithmic scaling of the BPM axis, the markers will appear at fixed distances from each other; this arrangement of markers may be shifted left or right.

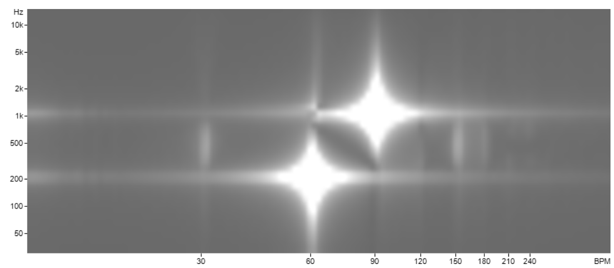

Fig. 6: Two sine waves of 200 and 1000 Hz are modulated by exponentiated sine waves oscillating at 60 and 90 BPM, respectively.

5. RESULTS

The system can separate modulated sine waves as shown in Figure 6. Here the analyzer displays a signal formed by the sum of two sine waves, each modulated by $\exp(\sin(2\pi f t))$, where f is a frequency in the rhythmic range. The corresponding two peaks are obvious. Small intermodulation effects due to the non-linearity of the level detectors can be discerned.

Note that a modulation by a function such as $1 + a\sin(2\pi f t)$ would lead to a non-sinusoidal level signal and thus to harmonics in the BPM analysis. Due to the logarithmic scaling of the axes, each peak in Figure 6 has an asymptotic profile of the form $\exp(-a|x - x_0| - b|y - y_0|)$, from which results the star-like shape. Only by using substantially more complex filters, one could obtain an elliptic, Gaussian shape $\exp(-a(x - x_0)^2 - b(y - y_0)^2)$.

Typical rock and pop music contains clear rhythmic pattern in snare and bass drums, hi-hat and bass guitar, see Figure 7, 8 and 9. These are immediately

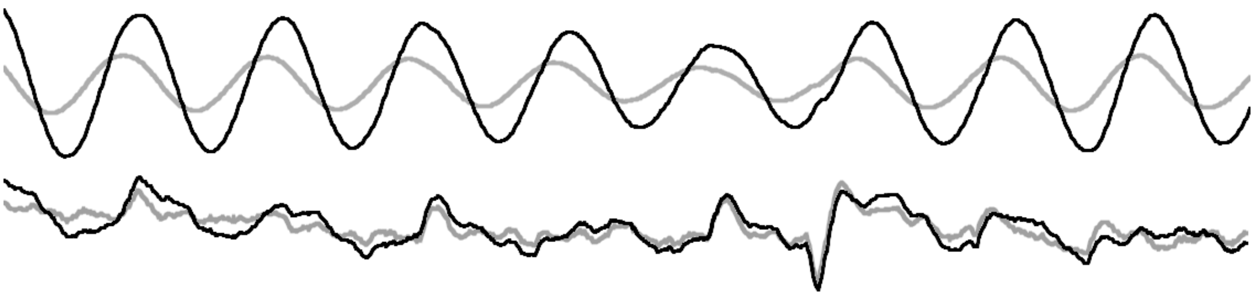

Fig. 5: . The user interface displays the level signals before and after the rhythmic equalization.

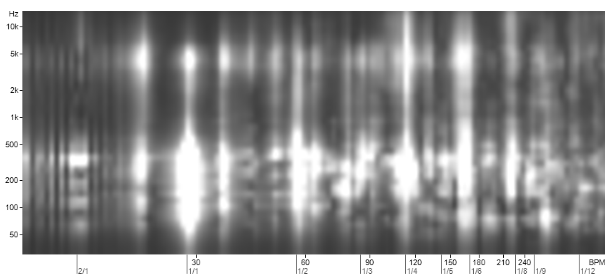

Fig. 7: Snare, bass drum, hi-hat and bass guitar form clear binary patters in this rock title.

accessible to a treatment in level and phase through the rhythmic equalizer presented here. Classical music, however, does not lend itself to be processed by the proposed system, since it is mostly based on non-percussive instruments and uses a flexible tempo. This is even true for classical dance music, see Figure 10.

6. CONCLUSION AND OUTLOOK

This paper presents methods to both analyze and edit music in real time employing an equalizer-style approach in the coupled domains of both audio and rhythmic frequencies. It may be used to apply subtle or drastic changes to pop/rock music and other styles with clearly marked and precise rhythms.

The proposed method can be extended to cover time-depended adjustments: The software could blend between different settings painted by the user;

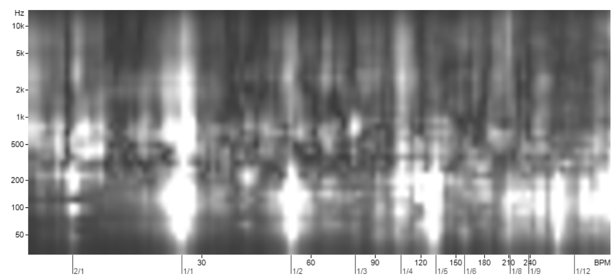

Fig. 8: This Latin pop music title contains a remarkable bass line which effectively is near 1/5 notes.

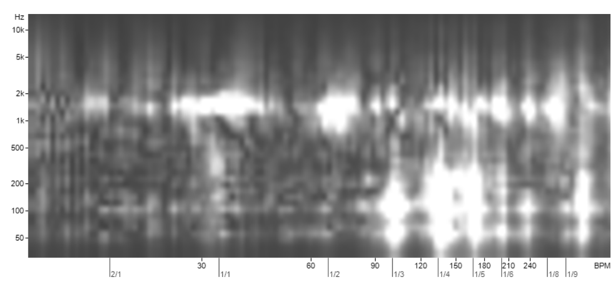

Fig. 9: The drums and bass of this up-tempo 5/4 jazz-rock title creates a clear BPM peak at five times the frequency of one bar.

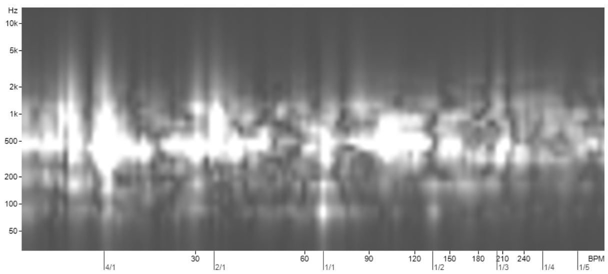

Fig. 10: In this waltz played by a classical orchestra, the overarching four-beat structure is more pronounced than the 3/4 rhythm.

it may even record and reproduce a temporal sequence of paint strokes. The IIR bandpass filters lead to small ripples in the phase and frequency responses. Even though this is not detrimental to the overall sound quality, the signal processing would become more accurate if one replaces the IIR filters in both the audio and the rhythmic domain with linear-phase FIR filters. Such filters for the lower BPM range would, however, introduce a substantial latency time.

7. ACKNOWLEDGMENTS

This work was funded by grant 1742B04 of the German Ministry of Education and Research (BMBF). The views and conclusions contained in this document are those of the author.

8. REFERENCES

[1] T. Dau, B. Kollmeier and A. Kohlrausch, "Modeling Auditory Processing of Amplitude Modulation. I. Detection and Masking with Narrow-Band Carriers," J. Acoust. Soc. Am. 102, pp. 2892–2905, 1997.

[2] E. Pampalk, S. Dixon and G. Widmer, "On the Evaluation of Perceptual Similarity Measures for Music," DAFX-03: Proceedings of the 6th Int. Conference on Digital Audio Effects, pp. 7–12, 2003.

[3] M. F. McKinney and J. Breebaart, "Features for Audio and Music Classification," ISMIR 2003: Proceedings of the 4th Int. Conference on Music Information Retrieval, pp. 151–158, 2003.

[4] T. Kinnunen, "Joint Acoustic-Modulation Frequency for Speaker Recognition," ICASSP 2006: Proceedings of the Int. Conference on Acoustics, Speech and Signal Processing, pp. I-665–I-668, 2006.