Research Article

Automatic Noise Gate Settings for Drum Recordings Containing Bleed from Secondary Sources

Michael Terrell, Joshua D. Reiss, and Mark Sandler

The Centre for Digital Music, School of Electronic Engineering and Computer Science, Queen Mary University of London, London E14NS, UK

Correspondence should be addressed to Michael Terrell, michael.terrell@eecs.qmul.ac.uk

An algorithm is presented which automatically sets the attack, release, threshold, and hold parameters of a noise gate applied to drum recordings which contain bleed from secondary sources. The gain parameter which controls the amount of attenuation applied when the gate is closed is retained, to allow the user to control the strength of the gate. The gate settings are found by minimising the artifacts introduced to the desirable component of the signal, whilst ensuring that the level of bleed is reduced by a certain amount. The algorithm is tested on kick drum recordings which contain bleed from hi-hats, snare drum, cymbals, and tom toms.

1. Introduction

Dynamic audio effects apply a control gain to the input signal. The gain applied is a nonlinear function of the level of the input signal (or a secondary signal). Dynamic effects are used to modify the amplitude envelope of a signal. They either compress or expand the dynamic range of a signal. A noise gate is an extreme expander. If the level of the signal entering the gate is below the gate threshold, an attenuation is applied. If the level of the signal is above the threshold the signal passes through unattenuated. The attack and release parameters control how quickly the gate opens and closes. As the name suggests, noise gates are used to reduce the level of noise in a signal. There are many audio applications, for example, noise gates are used to remove, breathing from vocal tracks, hum from distorted guitars, and bleed on drum tracks, particularly snare and kick drum tracks. The use of digital audio workstations (DAWs) for postproduction means that it is quick and easy to manually remove some sources of noise by silencing regions of an audio file. However, it is very time consuming to manually remove bleed from drum tracks so noise gates are still heavily used.

The reader is referred to [1] for a comprehensive review of digital audio effects (DAFx). In [2], a class of sound transformations called adaptive digital audio effects (A-DAFx) are defined. Adaptive effects extract features from a signal and use them to derive control parameters for sound transformations. Adaptive audio effects have existed for many years. Dynamic effects are simple examples of A-DAFx because the control gain applied is derived from the level of the input signal. Features can be extracted from the input signal, an external signal, or the output signal before being mapped to control parameters. These are referred to as *autoadaptive*, *external-adaptive,* and *feedback-adaptive* respectively. Cross-adaptive effects use two or more inputs; the features of which are used in combination to produce the control parameters for the sound transformation.

A-DAFx have been used for automatic mixing applications. Early work focused on audio for conferencing. An adaptive threshold gate is presented in [3]. This is an external adaptive effect. Ambient noise is picked up by a secondary microphone from which the level is extracted. The level of the noise is mapped to the threshold of a noise gate which is applied to the primary microphone. In [4], a direction sensitive gate is presented. This is a cross-adaptive effect. Each microphone unit contains two microphones. These face toward and away from the speaker. The level of the signals entering the microphones is extracted and

compared to determine the direction of the signal. The direction is mapped to an on/off switch which ensures that the microphone is only active if the sound source is in front of it.

Recent automatic mixing work has turned toward audio production. Perez-Gonzalez and Reiss [5–7] have presented A-DAFx for live audio production. A cross-adaptive effect which does automatic panning is presented in [5]. The automatic panner extracts spectral features from a number of channels, each of which corresponds to a different instrument. The spectral features are mapped to panning controls, subject to predefined priority rules. The objective is to separate spatially those instruments with similar frequency content. The work in [6] is used to reduce spectral masking of a target channel in a multichannel setup. This is a cross-adaptive effect. It extracts spectral features from each channel, and if a channel has a similar spectral content to the predefined target channel an attenuation is applied. Automatic fader control is demonstrated in [7]. This is a cross-adaptive effect. It extracts the loudness from each channel. Loudness is a perceptual feature, a function of level and spectral content. The loudness of each channel is compared to the average loudness of all channels and is mapped to fader controls. This mapping seeks to make the loudness of all channels equal.

In [7] the cross-adaptive effect is used to instantiate changes to the fader controls which seek to produce a predefined outcome: equal loudness in all channels. This can be viewed as a form of real-time optimization. There are a few examples of audio effect parameter automation, where the optimization is performed offline. Whilst these do not fit neatly into the A-DAFx structure, they still incorporate feature extraction and feature mapping. In [8], a method is presented which allows perceptual changes in equalization to be made to an audio signal. An example requirement is to make the signal sound brighter. This is a cross-adaptive effect. The spectral features of the input signal are extracted and are compared with a database of previously examined signals, to which perceptually classified equalization changes have been made. A nearest neighbour optimization is used to map the similarity in spectral features to relevant equalization settings. In [9], a method is presented which automatically sets the release and threshold of a noise gate applied to drum recordings. This work is expanded here. This is an autoadaptive effect. The distortion to the target signal and the residual noise are extracted from the input signal. An objective function is defined which is a weighted combination of these two features. The objective function is minimised subject to weighting parameter, mapping the features to the release and threshold.

Automatic audio effects for musical applications generally have a user input which takes subjective considerations into account. For example, [5] has a global panning width control and [6] has a maximum attenuation control. The panning values output by the automatic panner are scaled between the center, and the user-defined global panning width. The maximum attenuation control defines the maximum gain reduction that can be applied to channels in order to reduce masking with the target channel. If the use of an audio effect cannot be defined in a purely objective way, it is advisable to decouple subjective and objective elements when attempting to automate it. In the case of a noise gate this distinction can be made clearly. The objective is to reduce the amount of noise, so the gate should attenuate the signal when noise is prevalent and should not attenuate when the wanted signal is prevalent. The subjective element is the level of attenuation that should be applied.

2. Method

2.1. Noise Gates in Drum Recordings. A noise gate has five main parameters: threshold (T), attack (A), release (R), hold (H), and gain (G). Threshold and gain are measured in decibels, and attack, release, and hold are measured in seconds. The threshold is the level above which the signal will open the gate and below which it will not. The gain is the attenuation applied to the signal when the gate is closed. The attack is a time constant representing the speed at which the gate opens. The release is a time constant representing the speed at which the gate closes. The hold parameter defines the minimum time for which the gate must remain open. It prevents the gate from switching between states too quickly which can cause modulation artifacts.

A typical drum kit comprises kick drum, snare, hi-hats, cymbals, and any number of tom toms. An example microphone setup will include a kick drum microphone, a snare microphone (possibly two), a microphone for each tom tom, and a set of stereo-overheads to capture a natural mix of the entire kit. In some instances a hi-hat microphone will also be used. When mixing the recording, the overheads will be used as a starting point. The signals from the other microphones are mixed into this to provide emphasis on the main rhythmic components, that is, the kick, snare, and tom toms. Processing is applied to these signals to obtain the desired sound. Compression is invariably used on kick drum recordings. A compressor raises the level of low amplitude regions in the signal, relative to high amplitude regions which has the affect of amplifying the bleed. Noise gates are used to reduce (or remove) bleed from the signal before processing is applied.

Figure 1(a) shows an example kick drum recording containing bleed from secondary sources. Figure 1(b) shows the amplitude envelope of the kick drum contained within the recording, and Figures 1(c) and 1(d) show the amplitude envelope of bleed contained within the signal. The large and small spikes up to 1.875 seconds in Figure 1(c) are snare hits and the final two large spikes are tom-tom hits. Figure 1(d) has reduced limits on the y-axis. This figure shows the cymbal hit at 0 seconds, and hi-hat hits, for example, at 1.625 seconds. The amplitude of these parts of the bleed is very low and will have minimal affect on the gate settings. Components of the bleed signal which coincide with the kick drum cannot be removed by the gate (because it is opened by the kick drum). The snare hits coincide with the decay phase of the kick drum hits and so will have the biggest impact on the noise gate time constants. If the release time is short, the gate will be tightly closed before the snare hit, but the natural decay of the kick drum will be choked.
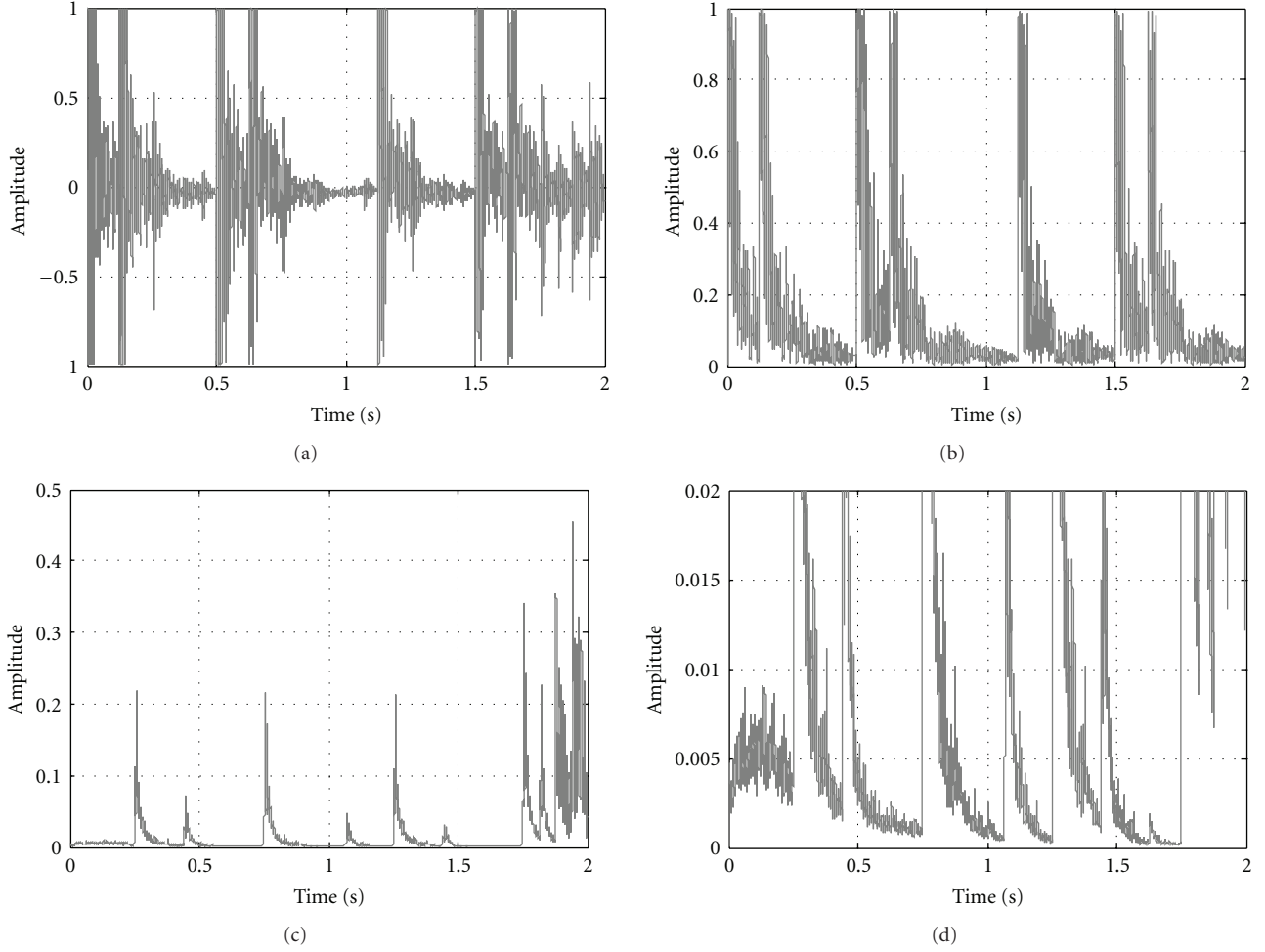
FIGURE 1: An example kick drum recording, (a) is a noisy microphone signal which includes kick drum and bleed, (b) shows the amplitude envelope of the kick drum contained within the noisy signal, and (c) and (d) show the amplitude envelope of the bleed contained within the noisy signal. Part (d) has reduced limits on the *y*-axis to show cymbals and hi-hats in the bleed signal.

If the release time is long the gate will remain partially open, and the snare hit will be audible to some extent, but the kick drum hit will be allowed to decay more naturally. If the threshold is below the peak amplitude of any part of the bleed signal, then the bleed will open the gate and will be audible. It is necessary to strike a balance between reducing the level of bleed and minimising distortion of the kick drum.

2.2. Audio Files, Artifacts, and Noise Reduction. Audio files representatives of a kick drum recording containing bleed from hi-hats, snare drum, cymbal, and tom toms are investigated. The audio is generated using the commercial software BFD2 from FXpansion. In this software the samples for each drum have been recorded with all microphones active so natural bleed is available. Test audio files are made by soloing the output of the kick drum microphone. Audio files are sequenced by the author. The kick drum signal which contains bleed is referred to as the noisy signal, $\mathbf{y}_n[n]$. This is a combination of the clean kick drum signal $\mathbf{y}_k[n]$ and the bleed signal $\mathbf{y}_b[n]$,

$$\mathbf{y}_n[n] = \mathbf{y}_k[n] + \mathbf{y}_b[n], \tag{1}$$

where $[n]$ is the sample index. $[n]$ will be dropped from this point onward for clarity. Time domain vectors are identified by lowercase, bold, typeface. Passing a signal through the noise gate will generate a gate function, $\mathbf{g}$. This vector contains the gain to be applied to each sample of the input signal. An example gate function is plotted in Figure 1(a). The gate function will generate distortion artifacts in the kick drum signal, D_A,

$$D_A = \frac{\left\| (1 - \mathbf{g})^T . * \mathbf{y}_k \right\|^2}{\|\mathbf{y}_k\|^2}, \tag{2}$$

and will reduce the bleed signal to a residual level, D_B,

$$D_B = \frac{\|\mathbf{g}^T . * \mathbf{y}_b\|^2}{\|\mathbf{y}_b\|^2}, \tag{3}$$

where $.*$ is the elementwise, vector multiplication operator. The signal to artifact ratio (SAR) and the reduction in the bleed level (δ_{bleed}) are given by

$$\text{SAR} = 20\log_{10}(D_A^{-1}),$$
$$\delta_{\text{bleed}} = 20\log_{10}(D_B). \qquad (4)$$

In [9] it is proposed that optimal noise gate settings should be found by minimising an objective function which is a weighted combination of the distortion artifacts D_A and the noise reduction D_N. The weighting parameter is then used to control the strength of the gate. The release and threshold are parameters in the objective function, but attack, gain, and hold are fixed. The attack is set to the minimum time of 1 ms, the gain to $-\infty$ dB, and the hold to a value that prevents distortion. A usable automatic gate requires these parameters to be included, in particular the gain setting, which if fixed at $-\infty$ dB will choke the kick drum sound severely. The implementation presented in this paper also includes the attack time and hold time as parameters in the objective function. The gain is used in place of the weighting parameter to control the strength of the gate. Rather than minimising an objective function which contains the distortion artifacts and the residual noise, the distortion artifacts are minimised (SAR is maximised), subject to the reduction in the bleed being greater than some threshold.

2.3. Approximating Distortion Artifacts and Noise Reduction. The distortion artifacts and noise reduction cannot be evaluated without separating the kick and bleed components of the signal. The human auditory system can do this instinctively. A human user will have prior knowledge of what the clean signal sounds like, that is, the user will know that the clean signal is a kick drum. This is replicated when automating the noise gate by inputting a single, clean, kick drum hit to the algorithm. In practice this could be obtained during a sound check, or could be taken from a database of kick drum samples.

The noisy signal is split into windows of quaver length. Each window is attributed to kick or bleed. The divisions within the noisy signal are made based on note onsets. Onsets are identified manually, but it is assumed that they could be identified exactly using an onset detection algorithm. The work in [10] is a benchmark paper on onset detection, and [11] contains a summary of drum transcription and source separation techniques. The spectral power of each window of the noisy signal is correlated with the spectral power of a region of the clean kick drum signal of equal length. If the correlation is above a predefined threshold, it is attributed to kick drum. The correlation is calculated as the scalar product of the normalised spectral powers. $\mathbf{X}_i$ is the spectral power of window i of the noisy signal, and $\mathbf{X}_c$ is the spectral power of the clean kick drum signal. The correlation is given by

$$\mathbf{c}_i = \left(\frac{\mathbf{X}_i}{\|\mathbf{X}_i\|} \right)^T \cdot \left(\frac{\mathbf{X}_c}{\|\mathbf{X}_c\|} \right), \qquad (5)$$

where $\mathbf{c}_i$ is the correlation of the spectral powers of window i of the noisy signal with the clean kick drum signal.

Windows of the noisy signal with a correlation greater than the threshold of 0.95 are assigned to kick drum. All other windows are assigned to bleed. An approximation of the clean signal is made by aligning a copy of the clean kick drum hit with the start of each window assigned to kick drum. This forms the synthesized clean signal $\mathbf{y}_z$, which is used in place of $\mathbf{y}_k$ in (2). The bleed is approximated by silencing all windows in the noisy signal which are attributed to the kick drum.

Figure 2 shows how the approximations to the kick and bleed components in the noisy signal are obtained. Figure 2(a) shows the noisy signal. It has been quantized with an eighth note quantization grid and windows are based on this spacing. Figure 2(d) of this figure shows the correlations between the spectral power of each window in the noisy signal with the spectral power of the clean kick drum hit. Marked on this figure is the correlation threshold of 0.95. All windows which contain a kick drum hit have a correlation above this threshold. Figures 2(b) and 2(c) show the synthesized kick drum signal, $\mathbf{y}_z$, and the approximate bleed signal, $\mathbf{y}_b$, respectively. The dotted lines on Figures 2(a) and 2(c) show the gate function $\mathbf{g}$, which is the gain applied by the gate as the noisy signal passes through it. The dotted line on Figure 1(b) shows the function $(1 - \mathbf{g})$. These are used to estimate the distortion artifacts and the residual noise as defined in (2) and (3).

2.4. The Noise Gate Optimization Algorithm. Common practice when using a noise gate to reduce bleed in drum tracks is to first set the gain to $-\infty$ dB. The threshold is then set as low as possible to allow the maximum amount of kick drum to pass through without allowing the gate to be opened by the bleed signal. The release is set as slow as possible whilst ensuring that the gate is closed before the onset of any bleed notes. For very fast tempos this may not be possible without introducing significant artifacts, in which case some bleed notes which occur close to the kick drum hit may be allowed to pass through. The implications of this in the automatic implementation will be discussed later. It is assumed that the gate must be closed for all bleed onsets. The attack is set to the fastest value which does not introduce any distortion artifacts. The hold time is continually adjusted to remove modulation artifacts caused by rapid opening and closing of the gate. During an interonset interval assigned to kick drum, the gate should go through one attack phase and one release phase only. The hold parameter should be as low as possible whilst maintaining this requirement. If it is too long it can affect the release phase of the gate. Once all other parameters have been set, the gain is adjusted subjectively to the desired level.

Figure 3 is a flowchart of the algorithm. The inputs on the left are constraints enforced at each stage. The inputs on the right are the parameter values at each stage. The signal is split into regions which contain kick drum and regions which contain bleed, as discussed in Section 2.3. An initial estimate of the threshold is found by maximising the SAR, subject to the constraint that the bleed level is reduced by at least 60 dB. This is identified by the parameter
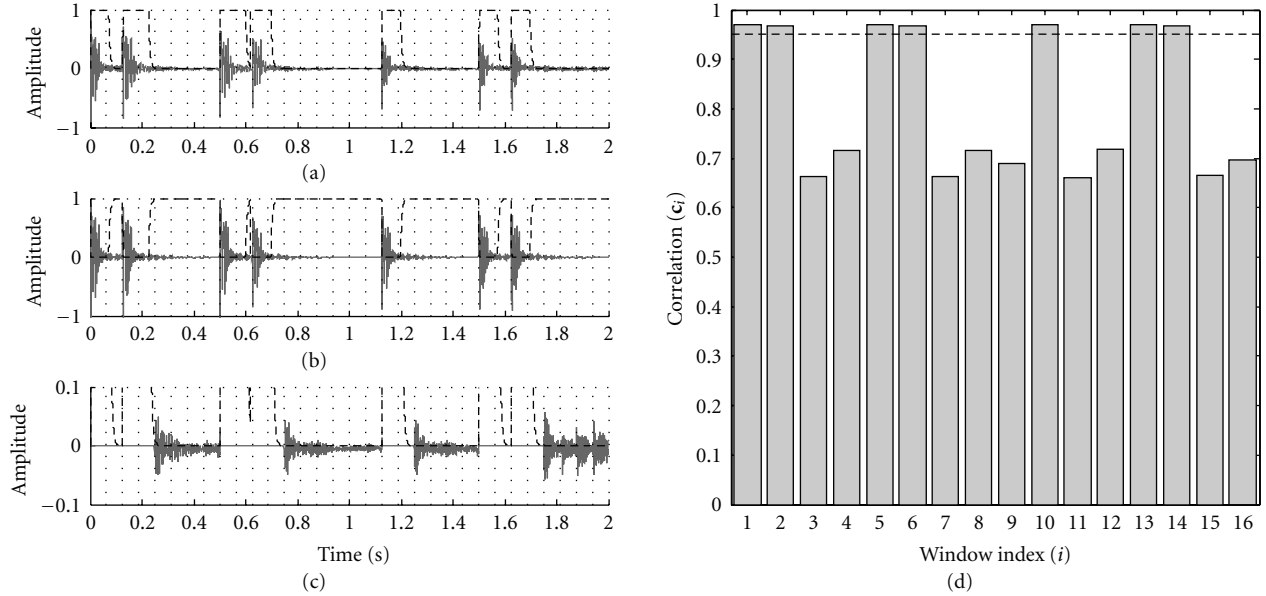
FIGURE 2: Approximations to the kick drum and bleed signals, (a) contains the noisy signal $\mathbf{y}_n$, (b) contains the synthesized clean kick drum signal $\mathbf{y}_z$, (c) contains the component of the signal attributed to bleed $\mathbf{y}_b$, and (d) shows the correlation of the spectral power of each window with the spectral power of the clean kick drum signal. The correlation threshold is identified by the dotted line.

δ_{bleed}, which is the minimum change in the bleed level after gating. The attack, release, and hold are set to their minimum values during the initial threshold estimate and the gain is set for full signal attenuation ($G = 0$ on a linear scale). This ensures that the threshold is set to the lowest feasible value. The minimum hold time is found which permits only one attack phase and one release phase for each kick drum window. These constraints are identified by parameters N_{attack} and N_{release} which correspond to the permitted number of attack and release phases, respectively. The other gate inputs are the minimum values of attack and release and the initial threshold estimate. The threshold estimate is required because the minimum hold time can vary significantly with threshold. The threshold is then recalculated using the updated hold parameter. Finally the attack and release are found by maximising the SAR, subject to the bleed reduction. Steepest descent gradient methods are used to minimise functions at each stage.

Breaking the algorithm into stages rather than defining a single objective function which contains all parameters has a significant advantage in this kind of optimization scheme. The major problems when using a single objective function are discontinuous regions in the solution space and regions of the solution space which have zero sensitivity with respect to small changes to the parameters. This is the case for all parameters when the threshold is close to zero (at which point the signal level is always above the threshold). By optimising each parameter in turn, and ensuring that the start point lies within a sensitive, continuous region at each stage, this problem is overcome. Alternative optimization methods which do not rely on gradient information could potentially be used.

3. Results

The algorithm is tested using a simple drum beat. The tempo of the beat is 120 bpm, the time signature is 4/4, and the kick hits lie on a 1/8 note quantization grid. There are some 1/16 note snare drum hits, but none of these occur immediately after a kick drum hit. This ensures that each kick drum window has a length of 1/8 note. The required bleed reduction is set to $\delta_{\text{bleed}} = -60$ dB, and the gain of the noise gate is set to $-\infty$ dB, that is, full attenuation. Figures 4(a) and 4(b) show the signal before and after gating, respectively. The gate function is plotted with a dashed line. It can be seen that the kick drum decay phase of the gated kick drum has been shortened, so that the signal level is approximately zero at the beginning of the region assigned to bleed, which occurs at 0.5 s. A user would now be free to adjust the gain parameter with the automated threshold, attack, release, and hold to change the strength of the gate.

The automatic noise gate algorithm is now investigated for a range of required bleed reductions, and for a range of noisy signals which contained different strengths of bleed. The strength of the bleed is measured relative to the test case described above, and includes bleed strengths of +0 dB, +2 dB, +4 dB, and +6 dB. Figures 5(a)–5(d) contain plots of the threshold, release, hold, and SAR, respectively. The attack has not been plotted because in all cases the algorithm set it to the minimum value of 1 ms.

Initial discussions are focused on the signal with a relative bleed strength of +0 dB. Figure 5(a) shows that the threshold has a stepped profile, and that it decreases as the required bleed reduction is decreased. Table 1 shows the peak levels extracted from each region of the noisy signal attributed to bleed. The overall peak level is -28 dB, which occurs in
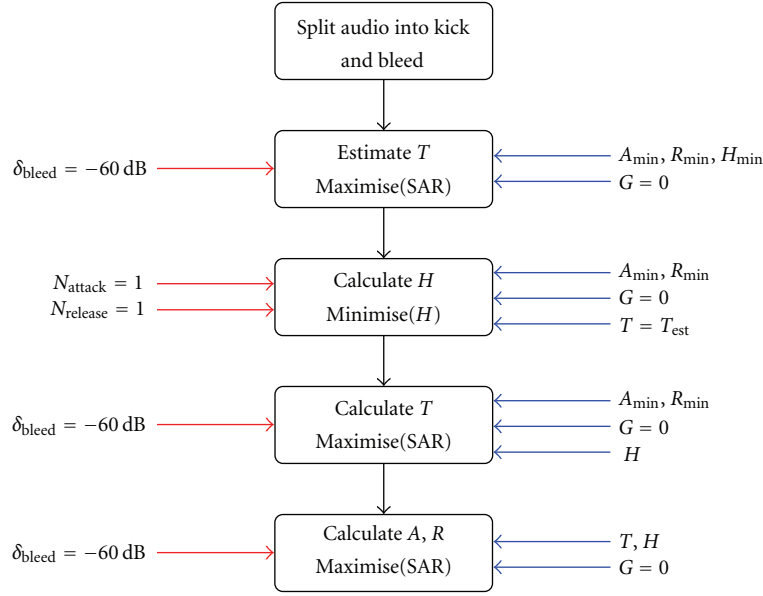
FIGURE 3: Automatic noise gate flow chart.

the final section and is due to the tom tom hits. Inspection of Figure 5(a) shows that the threshold is above this for $\delta_{\text{bleed}} < -10\,\text{dB}$, and so the bleed signal will not open the gate. Large reductions in bleed, for example, $\delta_{\text{bleed}} = -60\,\text{dB}$, result in thresholds which are higher than the peak level of the bleed by around 3 dB. This headroom is required to ensure that the gate has sufficient time to close during the release phase (which in calculating the threshold is set to the minimum value of 10 ms). As the required reduction in bleed becomes smaller, the gate does not need to be closed so tightly by the end of the release phase, which permits a lower threshold. The threshold follows a stepped profile because the bleed reduction is highly sensitive to small changes in the threshold. The threshold is set using the predetermined hold time and minimum attack and release times, as shown in Figure 3. Using these parameter values, a change in the threshold from $-25.89\,\text{dB}$, to $-22.56\,\text{dB}$ results in a change in δ_{bleed} from $-22.5\,\text{dB}$ to $-56.4\,\text{dB}$. With the tolerance used, there are no intermediate threshold values that will give a bleed reduction between $-22.5\,\text{dB}$ and $-56.4\,\text{dB}$. When the strength of the bleed is increased, a similar trend can be seen, but the difference between the threshold and the peak level of the bleed (shown in Table 1) gets progressively smaller. This is because with a higher strength of bleed, the absolute reduction in bleed to produce the same relative change is smaller, and the gate does not need to be closed so tightly by the end of the release phase.

For a fixed threshold the release time gradually increases as the required bleed reduction decreases. This is expected because the gate does not need to be closed so tightly by the start of the bleed window. Each step drop in threshold causes a sudden shortening of the time between the start of the release phase and the start of the following bleed window and so a step drop in release time is needed to produce the required bleed reduction.

TABLE 1: Peak signal level in the bleed regions identified by t_1 and t_2 for a range of relative bleed strengths.

t_1	t_2	0 dB	+2 dB	+4 dB	+6 dB
0.5	1	−29.1	−26.3	−25.6	−24.9
1.5	2.25	−29.1	−28.7	−28.2	−27.6
2.5	3	−29.3	−28.9	−28.4	−29.7
3.5	4	−28.0	−26.5	−24.5	−22.7

The hold time gives what appears to be the most unintuitive results. For signals with relative bleed strengths of +0 dB, +2 dB, and +4 dB, the hold time remains roughly constant at around 40 ms. The signal which has a bleed strength of +6 dB has a far lower hold time when the required bleed reduction is large, and shows a sudden increase in hold time when $\delta_{\text{bleed}} > -20\,\text{dB}$. The value of the hold time will depend on the degree to which the envelope of the kick drum signal is fluctuating about the threshold. If there are substantial fluctuations a longer hold time is required. The hold time is determined using the initial estimate of the threshold. Signals with different relative bleed strengths have different initial threshold estimates. Evidently for the signal with a bleed strength of +6 dB, there are minimal fluctuations in the envelope of the kick drum signal about the initial threshold estimate when the required bleed reduction is large. When the required bleed reduction is decreased, the initial threshold estimate is lower, and there are more fluctuations in the envelope of the kick drum signal about it. A longer hold time is therefore needed.

The SAR generally increases as the required reduction in bleed decreases. This is expected. A gentler gate causes less distortion in to kick drum signal. There are a few anomalous points where a decrease in the required bleed reduction is accompanied by an decrease in the SAR.
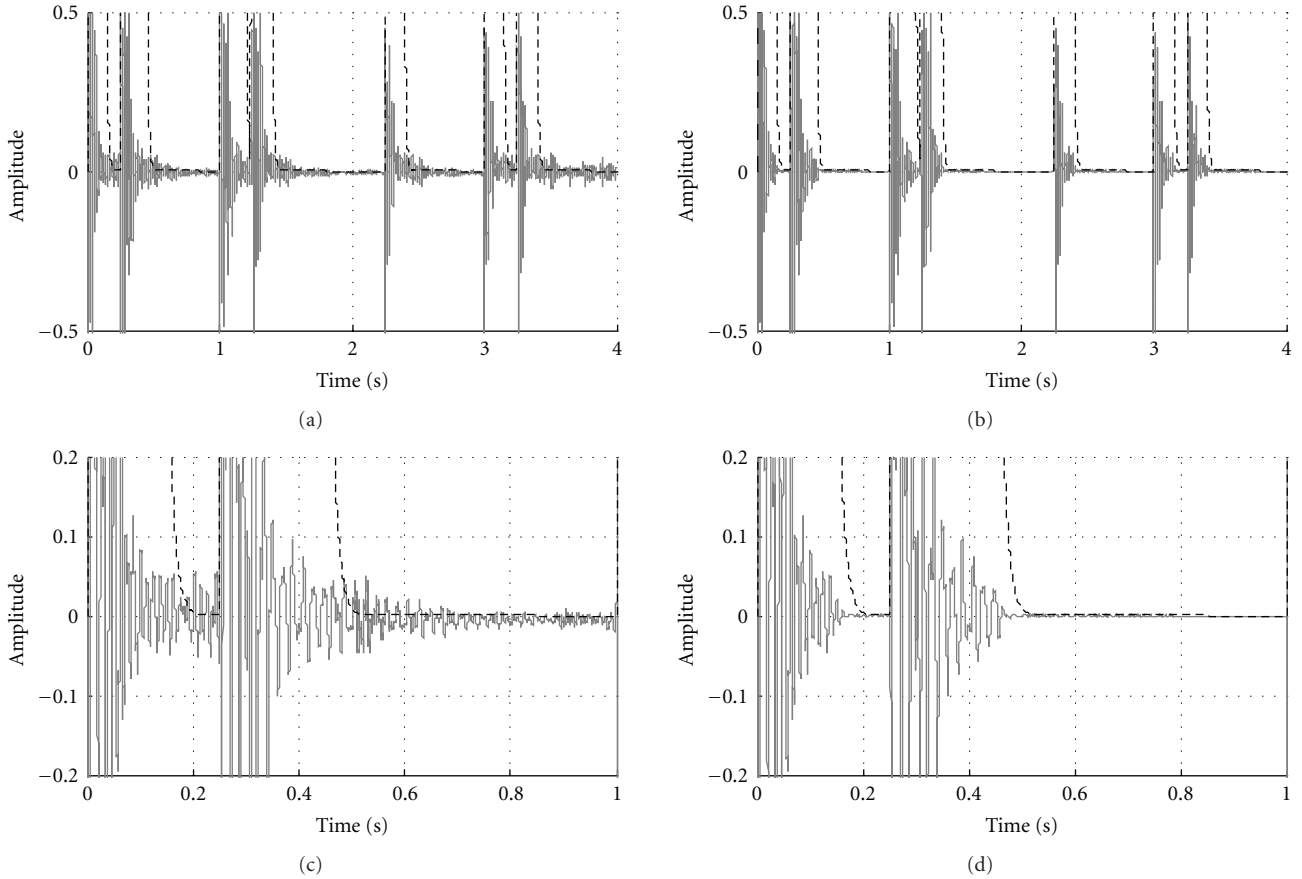
FIGURE 4: Kick drum recording before and after gating, (a) before gating, and (b) after gating, with $\delta_{bleed} = -60\,dB$.

These points coincide with step reductions in the threshold and release. It is suggested that in these transitional points a smoother change in the release and threshold may be required. This cannot be achieved with the algorithm in its current form because the threshold and release time are evaluated independently. It may be possible to include an additional, final stage which optimizes all of the parameters together.

4. Discussion

In designing the algorithm, manual use of a noise gate has been taken into account. It is the opinion of the author that by replicating the human thought process, the automated results should better approximate those obtained by a human user. Although formal evaluation has not been undertaken, informal testing has shown this to be the case.

The algorithm has been designed so that it is independent of the specific noise gate implementation. It would be easier to develop an algorithm if hidden aspects of the implementation, such as the transient filter properties, and the level detector, were known, but this would limit the use of the algorithm to a specific noise gate. This approach also ties in with the concept of replicating human operation because the parameters are set based only on the input and output of the gate and so much like with a human user, decisions are based purely on changes to the properties of the signal. It is the opinion of the author that this black box approach has most potential when considering commercial developments in the automation of any audio effect, as it allows the automation algorithm to be developed independently of the effect implementation (so long as the same parameters are available).

The algorithm presented divides the signal into a number of intervals based on the position of onsets. Problems will arise with drum recordings at high tempos and with high resolution quantization grids. In these cases it is likely that the kick drum regions will be very short, resulting in a choked kick drum sound after gating. A human operator would adjust the release to allow some bleed onsets which are close to the kick drum hit to pass through. This should be incorporated into the automatic gating algorithm. This could be done by defining a minimum kick drum window length, based on the amplitude envelope of the clean kick drum hit.

It is interesting to consider how the automatic noise gate presented in this paper fits into the A-DAFx framework. Most A-DAFx have a small analysis frame and update control parameters continuously, more or less in real time. This is particularly the case with established auto-adapative effects such as compressors. The algorithm presented here uses an audio segment of around 8 seconds, and takes 5–10 seconds to form and minimise the objective function. Despite this
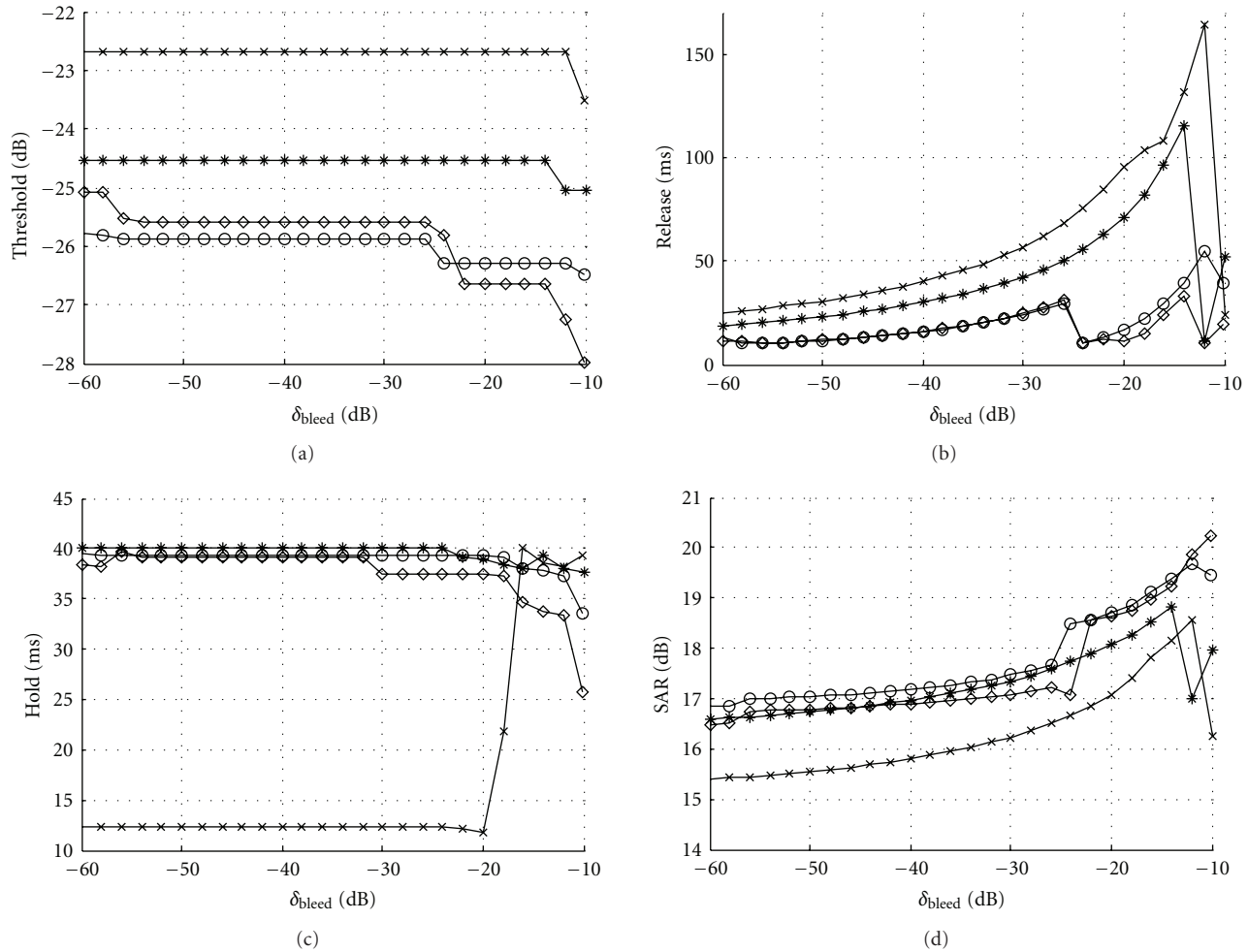
(a)

(b)

(c)

(d)

FIGURE 5: Noise gate parameter values after optimization, plotted against the required reduction in bleed (δ_{bleed} as defined in Figure 3). Part (a) shows threshold, (b) shows release time, (c) shows hold time and (d) shows SAR. Results are plotted for a number of relative bleed strengths identified by, $\diamond$: +0 dB, $\bigcirc$: +2 dB, $*$: +4 dB, $\times$: +6 dB.

lengthy time frame the algorithm could still be implemented within the A-DAFx framework. Large and sudden changes to noise gate parameters are undesirable, so an accumulative learning approach could be used as in [7].

Subjective evaluation has not yet been performed for this work. It would be useful to compare the values of the gate parameters output by the algorithm to those of an experienced engineer. This could be used to determine suitable reductions in SNR to be used in the algorithm, which may or may not be based on properties of the input signal.

5. Conclusions

An algorithm has been presented which automatically sets the threshold, release, attack, and hold parameters of a noise gate used on a kick drum recording that contains bleed from secondary sources. The parameters identified cause minimal distortion to the kick signal, whilst enforcing a predefined reduction in the level of the bleed signal. The gain parameter is not set automatically and is used to manually control the strength of the gate. The algorithm has been developed

independently from the noise gate implementation, and through consideration of the process followed by a human user. It has been tested for signals with varying levels of bleed, and varying amounts of bleed reduction. The gate settings found are intuitively correct, although as yet no subjective evaluation has been undertaken to compare them to expert users.

References

[1] U. Zolzer, *Digital Audio Effects*, John Wiley & Sons, New York, NY, USA, 2002.

[2] V. Verfaille, U. Zolzer, and D. Arfib, "Adaptive digital audio effects (A-DAFx): a new class of sound transformations," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1817–1831, 2006.

[3] D. Dugan, "Automatic microphone mixing," in *Proceedings of the AES 51st International Convention*, 1975.

[4] S. Julstrom and T. Tichy, "Direction-sensitive gating: a new approach to automatic mixing," in *Proceedings of the AES 73rd International Convention*, 1976.

[5] E. Perez-Gonzalez and J. Reiss, "Automatic mixing: live down-mixing stereo panner," in *Proceedings of the 10th International Conference on Digital Audio Effects (DAFX '07)*, 2007.

[6] E. Perez-Gonzalez and J. Reiss, "Improved control for selective minimization of masking using inter-channel dependancy effects," in *Proceedings of the 11th International Conference on Digital Audio Effects (DAFX '08)*, 2008.

[7] E. Perez-Gonzalez and J. Reiss, "Automatic gain and fader control for live mixing," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA '09)*, pp. 1–4, October 2009.

[8] D. Reed, "Perceptual assistant to do sound equalization," in *Proceedings of the International Conference on Intelligent User Interfaces (IUI '00)*, pp. 212–218, January 2000.

[9] M. Terrell and J. Reiss, "Automatic noise gate settings for multitrack drum recordings," in *Proceedings of the 12th International Conference on Digital Audio Effects (DAFX '09)*, September 2009.

[10] A. Klapuri, "Sound onset detection by appluing psychoacoustic knowledge," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '99)*, pp. 115–118, Phoenix, Ariz, USA, 1999.

[11] D. FitzGerald, *Automatic drum transcription and source separation*, Ph.D. thesis, Dublin Institute of Technology, 2004.