Dynamic Range Extension using Interleaved Gains

Paris Smaragdis, *Senior Member, IEEE,*
Adobe Systems Inc.
275 Grove St. Newton MA 02466, USA
paris@adobe.com

***Abstract*—We present a methodology to sample signals in such a way so as to avoid the effects of signal clipping due to a limited dynamic range. We do so by attenuating a selective subset of the data before it gets sampled, so that if clipping is detected after the sampling process we can easily estimate the missing samples using the non-clipped samples that were attenuated. We show that under sparsity assumptions it is possible to reconstruct the clipped samples and recover a satisfactory representation of the original signal. We provide an analysis of the side effects of this process and show that on average when sampling signals with highly varying or unknown gain, we can guarantee a significantly lower potential for signal distortion and noise.**

***Index Terms*—sampling, dynamic range, clipping, missing data, sparse reconstructions**

EDICS: AUD-SSEN

I. Background

Signal clipping is a well known problem in sampling theory and one that we encounter frequently with everyday use of devices such as audio recorders or digital cameras. Clipping is the effect of attempting to sample a signal whose values exceed the limits of the numerical representation in use, thereby losing information and obtaining a distorted sampled representation.

Traditionally, this problem is addressed by careful calibration of the sampling machinery in relation to the expected input, so that the likelihood of clipping is minimized. However this is a tedious and often unreliable process, which is rarely performed by non-specialists. In order to alleviate this problem, various automatic approaches have been devised. The most straightforward is that of manipulating the gain of the analog input signal before sampling. There is extensive work on signal processors that modulate incoming signals in such a manner so that their values are constrained within a specific dynamic range. These are known as compressors (or limiters) and have found extensive use in the audio production industry [1], [2]. In the speech community, such approaches are also known as automatic gain control (ACG) methods. Once past the digitization process, clipping can also be treated as a missing data problem where the clipped data are inferred using an appropriate signal model. There has been a significant amount of work on estimating missing data in this manner [3], [4], however these approaches are mostly concerned with obtaining a plausible reconstruction given a learned model. Because of that, they do not present a sampling methodology that can be employed for arbitrary signals, but are rather specialized solutions that require an appropriate signal model.

Recently there have been some interesting approaches in image capture technology that address this issue in the context of digital photography [5]. These approaches make use of modulation masks when sampling a signal and use that knowledge to reconstruct data that spans a larger range of values. In this paper, we expand on this idea and present an analysis of a similar approach as applied to audio data. We note that although the aforementioned approach works well for imaging, it results in very noticeable aliasing effects when used on audio data. Here, we introduce extra processing steps that guarantee more accurate sampling. Ultimately, we show how to guarantee increased resistance to clipping when sampling signals of unknown variance, and reconstruct clipped signals with minimal distortion.

II. Modulation for Clipping Avoidance

A. Sampling and Clipping

When sampling a signal, we perform two operations, one being the quantization of time and the other being the quantization of amplitudes. The problem we will address in this paper relates to the quantization of amplitudes, i.e., the conversion of a discrete-time continuous valued input $x[k]$ to a discrete and finite representation $x_{\mathrm{q}}[k]$. Most commonly today, we see the use of two's complement encoding for representing the input signal in a discrete manner. In this scheme, we define the quantization as being a rounding operation into 2^b discrete values, where b is the number of bits we use to represent the samples. Since this representation will be able to represent numbers valued from $\left[-2^{b-1}, 2^{b-1}-1\right]$, we will need to normalize $x[k]$ it so that it falls between these two values. We therefore define the following process as the quantization operation:

$$x_{\mathrm{q}}[k] = Q(\alpha x[k]) \tag{1}$$

where

$$\alpha = \frac{2^{b-1}-1}{\max |x[k]|} \tag{2}$$

and $Q(\cdot)$ is a function that rounds its input to the nearest integer. If the normalizing factor α is known in advance, then we can proceed by digitizing $x[k]$ and ensuring that its entire range will be properly represented by $x_{\mathrm{q}}[k]$. If however α is not known or it is overvalued, we run the risk of *clipping* the signal. This happens when the right hand side of (1) exceeds the range $[-2^{b-1},\ 2^{b-1}-1]$, and $x_{\mathrm{q}}[k]$ is assigned either of these two extreme values to represent an input that exceeds them. The result of this mishap when sampling audio signals is the introduction of a very harsh and unpleasant distortion, as well as loss of information. Alternatively we can select a small normalizing factor α such that no clipping occurs. In that

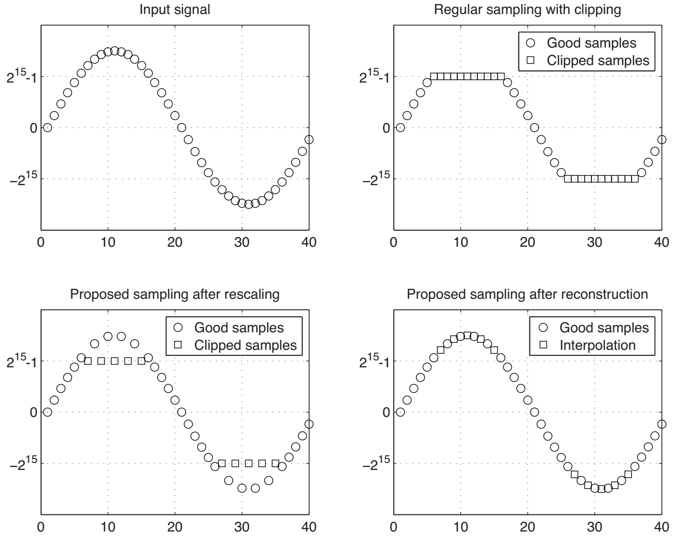

Figure 1. A basic example of the proposed sampling process. The top left figure shows the input signal, which surpasses the boundaries of the available sampling representation. The top right plot shows how this signal will ordinarily be sampled. The bottom left plot shows the reconstruction after we impose a gain mask, as described by (3), and the bottom right plot shows the result of interpolating to infer the samples that were lost due to clipping.

case, though, we run the risk of under-utilizing the dynamic range that the available bit depth provides, and in the process introduce extra quantization noise in addition to amplifying the potential pickup of ambient noise sources.

In the following section, we will outline a different approach to amplitude quantization that significantly offsets the effects of signal clipping to ones that are less noticeable and harsh.

B. Proposed Sampling Strategy

Consider the signal in the top left plot of Fig. 1. The signal is a sinusoid that varies in value from -50000 to 50000. Let us also assume that we need to sample this signal but are only provided with 16 bits of accuracy. The 16-bit samples will be able to express a signal range from -2^{15} to $2^{15}-1$, which is narrower than the available signal's range and will result in clipping as is illustrated in the top right of Fig. 1.

Let us consider then an alternative quantization approach. Before sampling an arbitrary input signal $x[k]$, we will impose the following modulation:

$$\begin{aligned} z[2k] &= x[2k] \\ z[2k+1] &= \frac{x[2k+1]}{2} \end{aligned} \tag{3}$$

which effectively scales every other incoming sample by a factor of 2. We then proceed by quantizing $z[k]$ to obtain $z_{\mathrm{q}}[k]$. The actual implementation of the scaling itself can be performed by a variety of methods, which are outside the scope of this paper. The simplest approach would be to multiplex the samples from two synchronized and offset ADC converters with different input gains. To reconstruct $x[k]$ from $z_{\mathrm{q}}[k]$, we need to undo the scaling by:

$$\begin{aligned} \hat{x}[2k] &= z_{\mathrm{q}}[2k] \\ \hat{x}[2k+1] &= 2z_{\mathrm{q}}[2k+1]. \end{aligned} \tag{4}$$

This conversion will of course require a representation with a higher dynamic range, otherwise we will see no benefit from the subsequent methodology. Upon doing so, we see that half of the large-magnitude samples, $\hat{x}[2k+1]$, have been sampled correctly whereas, the other half, $\hat{x}[2k]$, have been clipped. This is shown in the bottom left plot in Fig. 1. The clipped samples are easy to identify since they will be the ones assuming the extreme values of the chosen representation. Using interpolation, we can reconstruct the clipped samples of $\hat{x}[2k]$ from the available values of $\hat{x}[2k+1]$, as shown in the bottom right of the same figure. In order to guarantee accurate results when performing the interpolation, we will have to assume that the input signal is bandlimited and does not surpass half of the Nyquist rate. If it does so, then we risk the occurrence of aliasing. An additional concern is the introduction of some additional quantization noise from the extra scaling that is now imposed. These are the two main issues we need to address in this paper in order to guarantee proper sampling behavior.

C. Generalizations of the Basic Process

Obviously, the scaling imposed on $x[2k+1]$ does not need to be fixed to a factor of 2, but rather any value that appropriately expresses the expected uncertainty or fluctuation in the scaling of the input. In general, if we divide all $x[2k+1]$ values by c, then we can reconstruct signals that vary by at most c times the dynamic range of the available bit depth. Also, the rate at which we modulate the gain of the input doesn't have to be one out of every two samples, but rather any arbitrary number. We can therefore generalize this idea so that we use a sequence of normalizing factors (which we call a *gain mask*) on consecutive chunks of samples. This can be expressed as:

$$z[k] = C\left[k - n\lfloor\frac{k}{n}\rfloor\right]x[k] \tag{5}$$

where $\lfloor\cdot\rfloor$ denotes the floor operation and n is an integer denoting the length of the applied gain mask $C[k]$. In comparison, the approach shown above is a special case of this formulation where $C[0] = 1$, $C[1] = 1/2$, and $n = 2$. As shown later, using this formulation can achieve better control of the trade-offs between alleviating clipping and other artifacts that arise using this process. Another possible extension can use the fact that the scaling factor does not have to be constant across time. We can instead use a sampling strategy akin to automatic gain control which is characterized by:

$$\begin{aligned} z[2k] &= x[2k] \\ z[2k+1] &= \frac{x[2k+1]}{c[k]} \end{aligned} \tag{6}$$

where $c[k]$ can be modulated by estimating the smoothed amplitude of $x[k]$. In this case, when $x[k]$ is within the dynamic range limits, we can set $c[k]$ to 1, whereas when

$x[k]$ increases in magnitude we choose an appropriate $c[k]$ such that $x[k]/c[k]$ lies within the available dynamic range. This way, we do not risk corrupting the input signal when no clipping takes place, and we also ensure appropriate scaling in order to be able to reconstruct $x[k]$. The obvious trade-off with this approach, though, is that we need to store the variable $c[k]$, thereby complicating a hardware implementation of the sampling process. Likewise, the gain mask $C[k]$ in (5) can be either fixed or adaptively varying. For the remainder of this paper, we will mostly focus on the case shown in (3) for reasons of simplicity. We will however provide a treatment that is readily extensible to arbitrary gain masks and discuss some of their benefits.

III. Error Analysis

In this section, we will proceed with analyzing some of the side-effects of the proposed sampling approach. We identify two problems. The most important is that of aliasing that might occur during reconstruction. The other is increased quantization noise due to the scaling imposed on an already quantized input. We show that we can largely recover from the effects of aliasing by imposing a sparsity assumption on the signals, as well as by using the sign information from the clipped data. We also show that the effect of the quantization noise can be minimal, at least in comparison to the damage imposed by the clipping process.

A. Aliasing Artifacts

Aliasing artifacts can arise due to the fact that we discard some of the samples in the original signal and then perform reconstruction using interpolation. Let us start by analyzing a simple case. We start with the model described in the previous section by (3), where during the clipped regions we would lose every other sample. During reconstruction we interpolate to recover the missing samples of the signal. Because of the regularity of the missing information, there is no way to know if the available samples represent a frequency above half the Nyquist or below it. The process of interpolation would implicitly assume that we seek a bandlimited signal, which will result in "folding" the frequencies above half the Nyquist on to the lower half of the spectrum. This of course is a problem since it alters the spectral character of the input and introduces audible distortion. Dictating that the input should be bandlimited is not an acceptable option and although it is likely, we cannot hope that the high frequency content will be insignificant enough to not create an audible effect in the reconstruction. We therefore need to find a way to avoid aliasing during the reconstruction process.

We know from signal theory that avoidance of aliasing would ordinarily be impossible through simple interpolation, however we do have some additional information that can help us. Although we do not know the exact value of the clipped samples, we know their sign. During the clipping process, samples outside the sampling range are assigned the maximum or the minimum value of the representation, depending on their sign. These values provide us with the sign of the lost samples, which is still useful information. In addition to this, we also note that the frequency folding from the aliasing process results in a "busier" spectral profile due to the extra aliased frequencies appearing on top of the original ones. This means that the spectrum of that sound would be much more populated than otherwise. We can therefore call for the additional constraint that the spectral profile of the output has to be *sparse*. This constraint would result in looking for a reconstruction that will minimize the creation of new frequencies (such as the ones we see when aliasing). A straightforward way to impose this sparsity constraint is by ℓ_1-norm minimization of the frequency coefficients of the reconstruction.

All of the above constraints can be expressed jointly as the following linear program:

$$\begin{array}{lrcl} \text{minimize} & \mathbf{q}^T\mathbf{f} & & \\ \text{subject to} & \mathbf{K}\mathbf{F}^{-1}\mathbf{f} & = & \mathbf{K}(\hat{\mathbf{x}}\odot\mathbf{w}) \\ & \left|\mathbf{U}\mathbf{F}^{-1}\mathbf{f}\right| & \geq & \left|\mathbf{U}(\hat{\mathbf{x}}\odot\mathbf{w})\right| \\ & \mathbf{f}_i & \geq & 0 \end{array} \tag{7}$$

which is explained as follows. The first line defines the minimum ℓ_1-norm optimization in the frequency domain. The vector $\mathbf{q}$ is a weight vector, and $\mathbf{f}$ is a non-negative spectral representation of the sound we are reconstructing. The elements of $\mathbf{q}$ can all be set to 1, although in practice it is best to have its elements that correspond to high frequency coefficients of $\mathbf{f}$ to have slightly higher values, so that we obtain more sparsity in the higher frequency ranges. This effectively imposes a slight preference towards a $1/f$ spectral structure which we are likely to encounter in natural sounds.

The second line of the program is an equality constraint, which ensures that the unclipped samples should maintain their values. The matrix $\mathbf{K}$ is a diagonal matrix whose *i*th row diagonal element will be 1 if the *i*th element of the input sound is not clipped, and 0 otherwise. The matrix $\mathbf{F}^{-1}$ is an inverse spectral transform matrix that transforms the spectral coefficients of the non-negative vector $\mathbf{f}$ to the time domain. In this paper we used $\mathbf{F}^{-1} = \left[\mathbf{C}^T,\ -\mathbf{C}^T\right]$ where the $\mathbf{C}$ matrix is the DCT matrix. We repeated the DCT matrix using its positive and negative forms so that we can ensure that all the elements of $\mathbf{f}$ can be non-negative and still replicate any possible input. The vector $\hat{\mathbf{x}}$ is a short window of audio that we wish to reconstruct. It contains both re-normalized and clipped samples [e.g. the output from (4)]. In the experiments of this paper, $\hat{\mathbf{x}}$ was between 64 and 256 samples long. For long sounds, we used a sliding window method so that we reconstructed clipped samples one window at a time. The vector $\mathbf{w}$ is a window function that will help suppress spurious spectral elements in $\mathbf{f}$, so that the ℓ_1-norm minimization focuses on actual peaks and not sidelobes that appear due to poor frequency analysis. The operator $\odot$ signifies an element-wise (Hadamard) product, which imposes the windowing on the input signal. A Hann window wass used for most results in this paper.

The third line in this program imposes the constraint that the clipped samples should have values that exceed in magnitude the numerical limits of the sampling representation. Here, $\mathbf{U}$ is a diagonal matrix whose ith row diagonal element is 1 if the ith sample is clipped, and is 0 otherwise (such that $\mathbf{K}+\mathbf{U} = \mathbf{I}$).

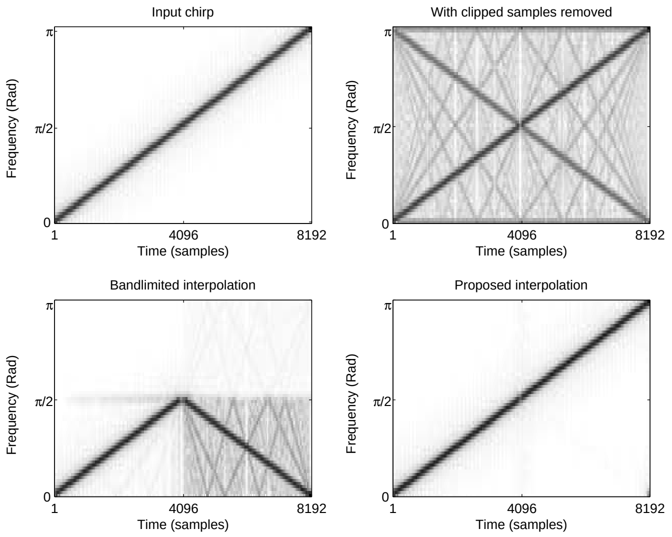

Figure 2. Reconstruction of a clipped chirp signal. The top left spectrogram shows the original input chirp signal. The top right plot shows the spectrogram of the chirp signal with the clipped samples removed. Note how this operation results in severe aliasing. The bottom left plot shows the result of replacing the clipped samples with values derived from bandlimited interpolation and the bottom right plot shows the reconstruction using our proposed method. It is easy to see that our reconstruction does a good job at bypassing aliasing and bandwidth limiting problems that we would otherwise observe.

Finally, the fourth line of the linear program dictates that the spectral representation that we use is non-negative, which is required in order to ensure the ℓ_1-norm minimization.

These equations constitute a straightforward linear programming problem whose solution has been covered extensively in the linear programming literature [6]–[8]. In this particular application, the data do not present any particular numerical challenge and we can estimate $\mathbf{f}$ efficiently using any off-the-shelf solver. To summarize the above program in more intuitive terms, what we try to do is to find a sparse spectral feature vector that in the time domain will result in an output that is equal to the known samples, and magnitude bounded from below by the clipped samples.

To demonstrate the use of this optimization, let us consider a simple illustrative problem. The signal we will use is a 8192-sample chirp signal, sweeping from zero frequency to Nyquist. The spectrogram of this signal is shown in the upper left plot of Fig. 2. The extreme values of this waveform are -2 and 2 and the clipping points were set to -1 and 1. Using the sampling process from (3) results in a little more than a third of the chirp samples getting clipped. Once we remove the clipped samples, we observe strong aliasing effects, which are shown in the upper right plot of the same figure. If we don't use the available sign information and naively perform bandlimited, interpolation, in order to reconstruct the clipped samples, we will force the latter half of the chirp to be reflected across the half-Nyquist frequency (bottom left plot). If we use the proposed reconstruction process, the aliased solution is not plausible and an accurate reconstruction is obtained despite the heavy loss of information (bottom right plot).

The chirp experiment is of course contrived and can only serve as an introductory illustrative example. So let us present a more realistic experiment on real audio data. Consider the speech signal shown in the top left plot of Fig. 3, a recording of a female speaker saying "a small boy". The peak of this signal is twice as large as the clipping threshold, which resulted in clipping with strong audible artifacts. The clipped samples are concentrated around the "a" in "small" and the "o" in "boy", which were the only regions that contained any clipped samples. Within these regions, the percentage of clipped samples was between 1% and 12% per window of 128 samples (which was the size of the sliding window used for reconstruction). Upon detecting the clipped samples, we used a few different ways to infer the missing information. In the top right plot we show the reconstruction using an autoregressive model trained on speech, in the bottom left we use bandlimited interpolation, and in the bottom right plot we use our method. For the AR model, we can see a significant amount of aliasing during the reconstruction of the clipped areas. We achieved qualitatively similar results when using simple polynomial interpolation instead. In the bandlimited interpolation case, it comes as no surprise that the upper half of the frequencies is suppressed around the areas where we have a high density of missing samples. The proposed reconstruction is a very satisfactory result, which exhibits no detectable aliasing or significant distortion of the input (either visually or in listening tests).

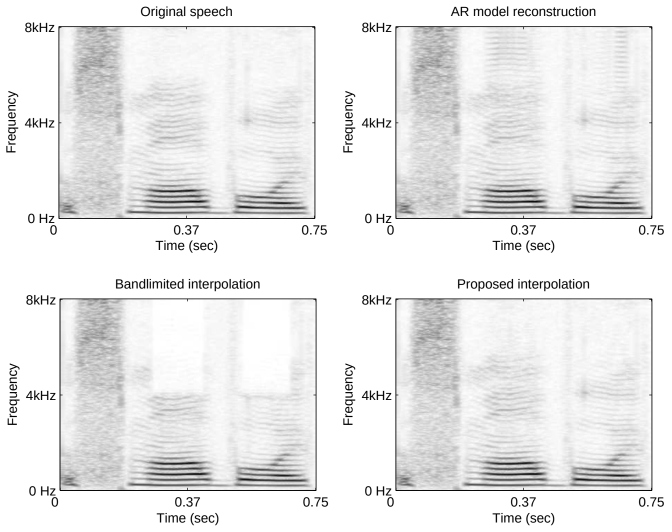

Figure 3. Reconstruction of a speech signal. The top left plot shows the spectrogram of the original speech signal. The top right plot shows its spectrogram after an AR model reconstruction where it is easy to see the aliasing effect during the loudest regions. The bottom left plot shows the results of a bandlimited interpolation where the upper frequencies are being suppressed. The bottom right plot shows the proposed approach where we can resynthesize without significant aliasing or loss of high frequencies.

B. Quantization Noise

Now let us focus on the case where no clipping takes place. These are cases where the proposed method is redundant and straightforward sampling would properly represent any input signal. The only artifact that our method will produce is that of excess quantization noise due to the extra normalization of some of the samples.

We start with the example where we scale every second sample by a constant factor c, i.e:

$$\begin{aligned} z[2k] &= x[2k] \\ z[2k+1] &= cx[2k+1]. \end{aligned} \tag{8}$$

We also assume that the sampling taking place is done with b bits of precision. If we assume that the quantization noise and $x[k]$ are uniformly distributed across all possible amplitude values, then ordinary sampling at b bits will result in $-10\log_{10}(2^{-2b})$ dB of signal-to-noise ratio (SNR) due to quantization. In our sampling scheme, half of the samples will have an SNR of $-10\log_{10}(2^{-2b})$ dB and the other half an SNR of $-10\log_{10}(c^{-2}2^{-2b})$ dB, which is averaged to yield the overall SNR. In this case, this will equal $-10\log_{10}\left(\frac{1}{2}(2^{-2b}+c^{-2}2^{-2b})\right)$. In the case of an arbitrary length gain mask $C[t]$, this extends to:

$$\mathrm{SNR}(b,c) = -10\log_{10}\left(\frac{1}{n}\sum_{i=1}^{n} c[i]^{-2}2^{-2b}\right). \tag{9}$$

If we have a time-varying gain $C[k]$ then the predicted SNR will be dynamically changing in time according to the above equation.

Simulations of the quantization noise effect when sampling white noise are shown in Fig. 4. In both cases we assume that we sample with 16 bits of precision. The left plot shows the case where the gain mask is $C[k] = [1, {}^1\!/\!_2]$. This means that the even samples will be effectively sampled at 15 bits. We show the measured spectrum of the quantization noise and compare it with the spectra of the 16-bit and 15-bit sampling noise. The noise under this sampling scheme is the average of that between the two implied bit precisions. One more example takes place in the right panel of Fig. 4. In this case we show the noise levels when we use the gain mask $C[k] = [1,\ {}^1\!/\!_2,\ {}^1\!/\!_3]$. This time, the implied quantization for each triplet of samples is 16, 15, and 14.4 bits, respectively. Once again we see that the overall quantization noise is the average of the used bit accuracies.

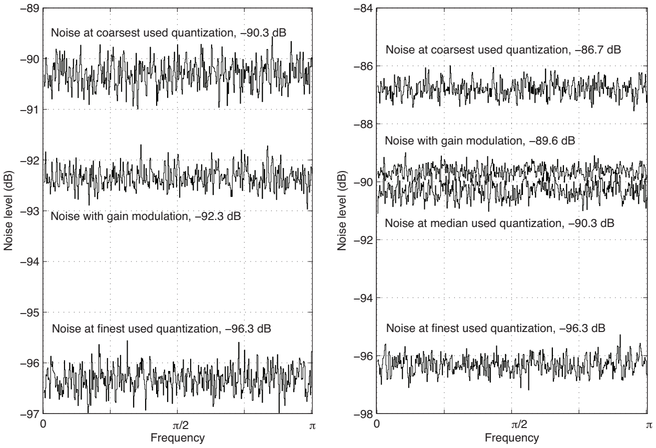

Figure 4. Quantization noise spectra when performing gain modulation. In the case shown in the left panel, we scale all odd samples by a factor of 2, which results in one less bit of precision for these samples. Sampling is done at 16 bits. The top trace shows the measured quantization noise of the odd samples, which is also the level expected for 15-bit sampling. The bottom trace displays the noise for the even samples, which is also the expected noise for 16-bit sampling. The overall noise level for the gain modulated signal is the middle trace. This is the linear average of the other two traces (note the log scaling in this figure). Likewise, on the right, we show the equivalent results for a gain mask $C[k] = [1,\ {}^1\!/\!_2,\ {}^1\!/\!_3]$. The top line is the noise for the third out of each triplet of samples, which gets effectively sampled at 14.4 bits. The bottom and third traces show the respective noise for the other two samples (sampled at 16 and 15 bits).

IV. Comparison to clipping

The ultimate goal of this paper is to recover an otherwise clipped signal with minimal distortion artifacts. To see how this approach compares to an ordinarily sampled signal we consider some examples and attempt to quantify the results.

Let us consider the case in (3) again. This time the input is 1536 samples long and contains 10π periods of a sinusoid (the number of periods was selected to be an irrational number, in order to minimize periodicity effects in the noise measurements). Sampling takes place at 16 bits of precision. Suppose that the sinusoid's extrema are twice as large in magnitude than what the available bit precision allows. This means that if we perform straightforward sampling, roughly 66% of the samples will clip. Using variable gain sampling, we only observe clipping in half of these samples, due to scaling half of them before quantization.

Let us investigate what happens to the spectral characteristics of the aforementioned sinusoid when performing these operations. To compute the power spectra we used a Hamming window and a frame size of 512 samples. In the case of straightforward sampling, we observe severe distortion, resulting in an SNR of 7.6 dB, and a significant presence of new harmonics in the signal. The spectrum of the extra noise added into the signal due to clipping, as compared to the input's spectrum, is shown in the top plot of Fig. 5. In comparison, the spectrum of the introduced noise due to our reconstruction results in an SNR of about 99 dB which is inaudible and roughly equal to the noise floor due to quantization. The spectrum of the introduced noise as compared to the that of the input is shown in the bottom plot of Fig. 5.

Now let us consider a real-world signal and illustrate the advantages of this method in a more realistic scenario. In Fig. 6 we display the results for a speech recording sampled at 48 kHz, where we contrast its power spectrum with that of the introduced noise due to clipping and the use of various gain masks. The top plot shows the input waveform in gray, and the limits of the numerical representation we used by the dashed lines. The input's peak value surpasses the sampling representation by a factor of 10, which results in clipping 28% of the samples. The subsequent series of plots show the spectra of the introduced noise using various gain masks, as compared to the spectrum of the input. One can easily see that the noise that is introduced when we don't use a gain mask is as strong as the input signal (SNR of about 0.2 dB), whereas increasingly more aggressive masking improves the SNR up to 44 dB. The audible effects of this noise are virtually imperceptible for most of the best performing gain masks. By comparison, if we knew *a priori* the extreme values of the

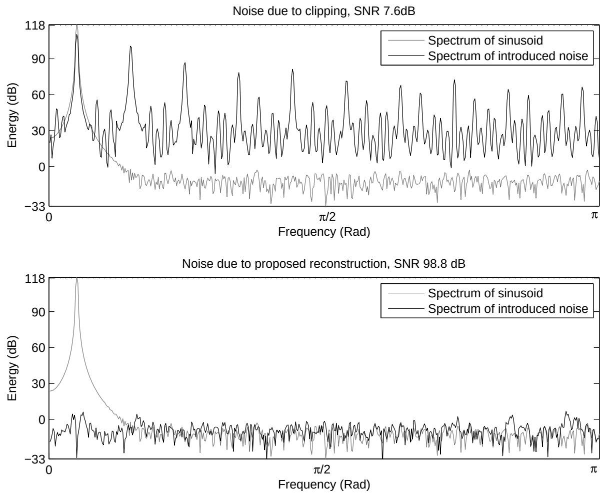

Figure 5. A comparison of the resulting noise spectra when sampling a sinusoid that exceeds the available dynamic range in a traditional manner versus the proposed approach. The dark trace of the top plot shows the power spectrum of the quantization noise due to clipping. In comparison, the grey trace shows the actual spectrum of the sinusoid after quantization but without any clipping. The bottom plot shows the same measures using our proposed method. The resulting noise pattern is significantly reduced and practically inaudible.

input signal and used the appropriate scaling to avoid clipping the quantization noise would rise up to about -76 dB. We therefore attain a noise performance within 32 dB of the limit of quantization noise with proper scaling, but by doing so we can capture signals with an unexpected 10-fold (20 dB) increase in dynamic range. It should also be noted that the kind of noise that we measure with this process is concentrated in the high-energy portions, whereas the low-energy portions exhibit less noise as compared with optimally scaled sampling. This creates a noise pattern that is less perceptible than the usual quantization noise. An illustration of this is shown in Fig. 7.

We repeated the above experiment for multiple speech and music waveforms and, in addition to the SNR, we also measured the Perceptual Evaluation of Audio Quality (PEAQ) scores [9] of the sampled representations in order to obtain an indication of the perceived degradation. Using ordinary sampling, 27% of the samples were clipped for the speech signals, and 41% for the music signals. The averaged results for that data are shown in Fig. 8. We clearly see that our approach increases the SNR dramatically with more aggressive gain masks. The PEAQ scores also indicate that the sampling noise ranges from “highly annoying” in the case where we have clipping, to “imperceptible” in the case where nearly correct gain scaling has been applied. This has also been verified in informal listening tests. For reference a 160-kbit/s MP3 encoding of these sounds results in a PEAQ index of about -0.5.

Another interesting point is the difference between speech and music results. Speech, having higher kurtosis than music, has a smaller ratio of samples near the extreme amplitude values. This implies that clipping and its subsequent reconstruction occurs less often than it does in music, which has a

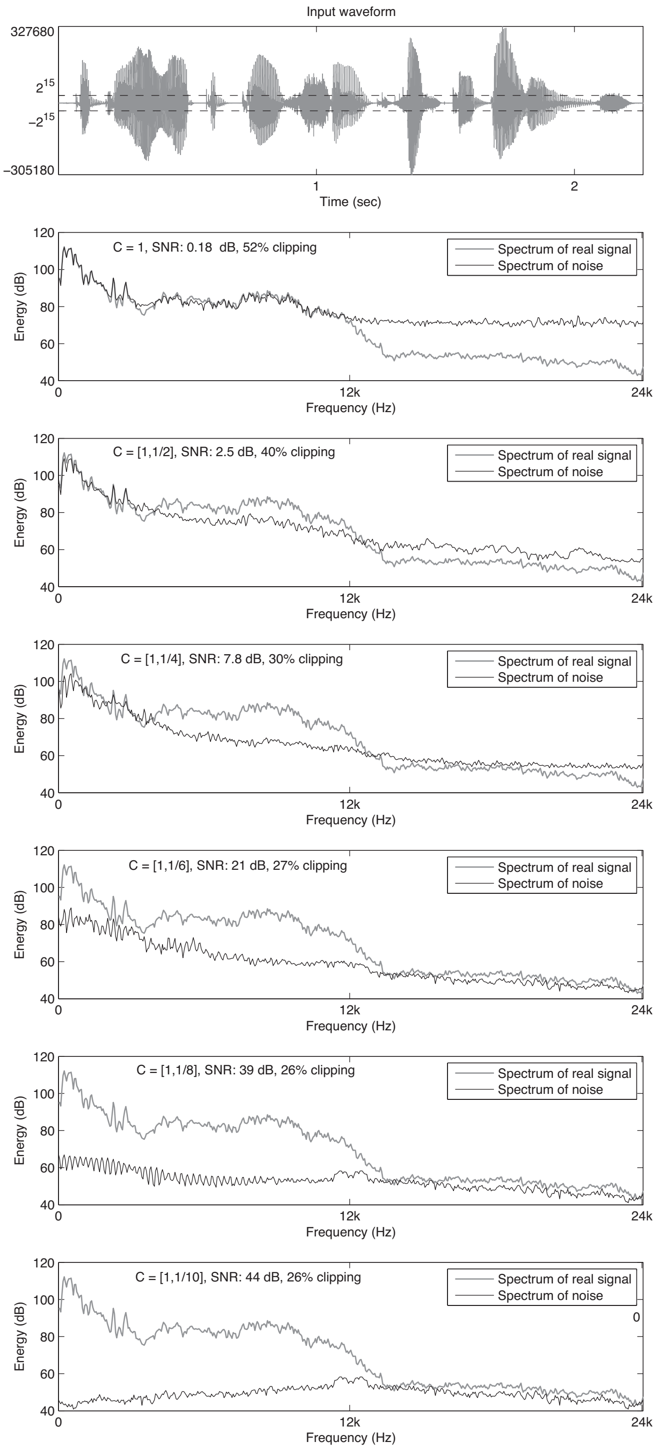

Figure 6. Sampling error spectra for a speech signal using various gain masks, as compared to clipping and normal sampling. The topmost plot displays the input waveform (in gray) and the limits of the sampling representation we used by dashed lines. At its peak the waveform exceeds these limits by a factor of 10. The second plot from above displays the power spectrum of the noise introduced due to clipping. For reference, the spectrum of the actual input is also shown by the gray line. Subsequent plots display the spectra of the sampling noise for a variety of gain masks. The text in the plots shows the gain mask that was used, the resulting SNR in dB, and the average percentage of clipped samples per analysis window using only windows where at least one sample was clipped and reconstruction was necessary.

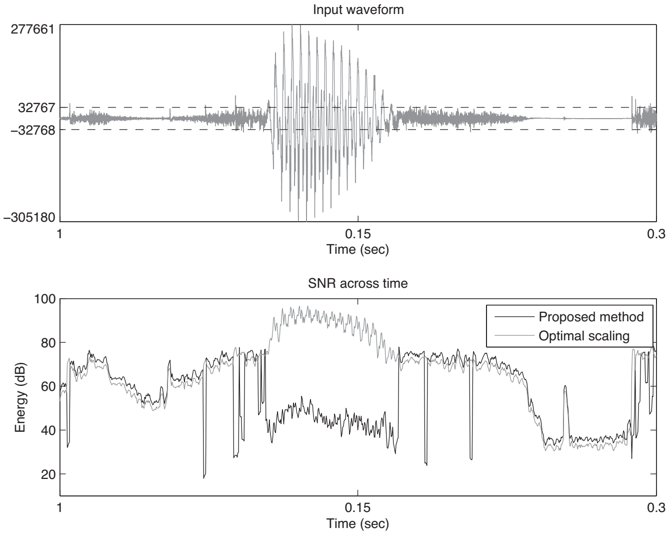

Figure 7. SNR measurements of a speech signal over time. The top plot shows the input waveform with the clipping thresholds shown by the dashed lines. The bottom plot compares the SNR measurements between the optimal scaling, which would guarantee no clipping (grey trace), and the proposed method (black trace). Note how our method samples the low energy parts with less quantization noise and introduces noise only in the loudest sections.

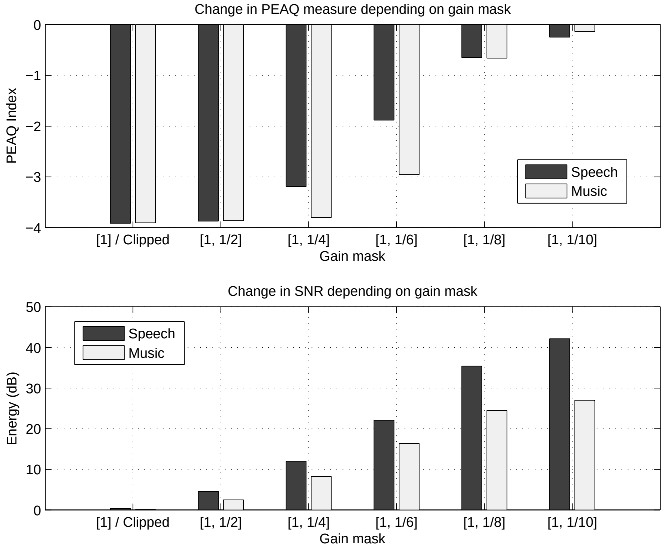

Figure 8. Averaged measurements of PEAQ (top) and SNR (bottom) in multiple speech and music examples. The labels on the x axis denote the gain mask that was used. The leftmost case, where the mask was $C[k] = 1$, is equivalent to normal sampling without any gain modulation. The input signals were scaled so that their maximum value would be 10 times the maximum sample in the used representation. Therefore the rightmost bars denote the case where, at worst, only one sample out of every two is clipped. The cases on the left are the worst-case scenarios.

more uniform distribution of amplitude values. This results in smaller SNR gains for the music signal. On the other hand, the PEAQ scores on music sounds are more favorable since the resulting distortion is not as prominent within a busy ensemble of many wideband sounds.

V. Discussion

So far we only considered simple gain masks in order to make the presentation more accessible. Let us now discuss some of the more interesting directions that we can take using this approach.

As we pointed out earlier, the size and composition of the gain masks can be of arbitrary length. Selecting an appropriate mask in such cases can be an involved procedure. Ideally we would like to ensure that the clipped samples within a mask will be as few as possible. This implies that the most cautious selection for a mask would be one where most of the gains are small, with only a few taking large values. This will result in having to reconstruct a minimal number of samples and a small risk potential of reconstruction error. On the other hand, this will result in increasing the average quantization noise of the samples, since most inputs will be scaled down. A more aggressive selection would be the opposite approach of using many large gains and very few small gains. This will result in having to reconstruct more missing data, but with little impact in the effective number of bits used to represent the samples.

A choice in the spectrum between these options can be based on knowledge of the kind of input sound. A highly kurtotic signal, like speech, can benefit from an aggressive choice of a gain mask, whereas a more uniform signal, like a popular music recording, would not. Since the qualitative tradeoffs between clipping and our reconstruction are not easy to analyse or quantify, making an automatic decision on the optimal gain mask based on signal statistics is not easy, and at this point we can only rely in a mask selection based on heuristics.

In either of these cases, it is prudent to carefully arrange the gain mask elements in order to help the reconstruction process. For example, it is best to use a mask that can result in smaller sections of contiguous clipped data, such as $[1, 1/2, 1, 1/3]$, as opposed to $[1, 1, 1/2, 1/3]$. This makes the reconstruction process simpler and also constructs an aliasing pattern that is easier to resolve (the second mask can result in “double” aliasing as opposed to the regular aliasing we would get with the first one).

When thinking in terms of aliasing effects, another factor that becomes relevant is the sampling rate we choose. One can consider the extreme case where we use a two-sample gain mask and oversample the (bandlimited) input by twice the required sampling rate. This means that even if we lose one in two samples to clipping, we can still reconstruct the signal without loss of information. This is because we avoid aliasing, which mixes the real and folded frequencies by increasing the sampling rate. The generalization of this statement is that as long as we oversample a bandlimited input by the length of the gain mask, then there will be no serious aliasing issue. Of course this is not particularly practical and is essentially equivalent to sampling the same signal while multiplexing inputs with multiple gains. However, even oversampling by a fractional amount will result in minimizing the aliasing overlap and can lend to better reconstruction. In cursory simulations, oversampling by as little as 10% resulted in an SNR increase of 5 dB, whereas oversampling by 20% resulted in a 9-dB increase.

An alternative approach, which can average out the mask design options, is the use of a completely random gain mask. Preliminary simulations show that this can result in an improved SNR and average out the tradeoffs in the aforementioned options. However this type of modulation can significantly complicate a working hardware implementation and is beyond the scope of this paper.

Finally, it is possible to use automatic gain control to adjust some of the elements of the gain mask so as to ensure nearly optimally masking at all times. The advantage of this approach will be that although we perform a sort of gain control, we will not be distorting the amplitude of the reconstruction, but instead use this scaling to infer the clipped samples.

VI. Conclusions

In this paper, we have shown a sampling methodology that can result in a high tolerance to clipping when sampling signals with extreme gain fluctuations. The sampling approach can be easily implemented in hardware since it only involves the design of a gain mask stage. The remainder of the sampling apparatus remains the same as regular sampling systems. Implementation of the waveform restoration in hardware is a much more complex endeavor, however it is not required and in fact to be avoided. Aside from complications in implementing a linear program solver in hardware, restoring a clipped waveform would necessitate a change in numerical format to one that is able to accurately represent the reconstruction. This only postpones the problem of guessing the input's extrema from the gain stage to the storage stage. Even if we guess right, we would have to use considerably more storage space to cover the expanded dynamic range, which is not a practical decision. Instead we can keep the gain-modulated and clipped waveform in the bit width that the sampling hardware already uses and then perform the reconstruction in software using floating point when we need access to the actual waveform. This would be akin to sampling the signal in a compressed format and then having to decode it when waveform access is needed. This way we can keep the hardware implementation very simple and provide an efficient storage format for much larger dynamic ranges than otherwise.

We should also note that the main benefit of this approach is not higher fidelity sampling, but rather the design of a process that can tolerate gross miscalculations in gain settings, or deal with wide energy fluctuations in a signal, and still produce an acceptable sampling performance. These are situations that are commonplace with consumer products such as handheld audio and video devices, which often produce clipping in recordings, but also in studio situations when using close miking techniques on dynamic sources, such as drums and voice, in order to reduce the presence of reverberation and ambient noise. Alternative sampling options in these cases are either the use of automatic gain control or re-recording with adjusted gains. The former approach will result in severe misrepresentation of the signal, which depending on the intended application of the recording, might be undesirable. The latter approach requires extra work and time in addition to requiring a reliably repeatable source, and offers no guarantee of success when repeating the recording process. Given these two constraints, the alternative we present is a valuable approach to recording reliably under uncertain gain conditions.

References

[1] J. M. Woram. *Sound Recording Handbook*, SAMS Publishing, 1992.

[2] F. Wylie. “Audio compression technologies,” in *NAB Engineering Handbook*, J. C. Whitaker (ed.), Washington D.C.; National Association of Broadcasters, 1998.

[3] R. Veldhuis. *Restoration of Lost Samples in Digital Signals*. Upper Saddle River, NJ: Prentice-Hall, 1990.

[4] S. J. Godsil and P. J. W. Rayner. *Digital Audio Restoration - A Statistical Model Based Approach*, Springer-Verlag London Limited, 1998.

[5] S. K. Nayar and T. Mitsunaga. “High dynamic range imaging: spatially varying pixel exposures,” in Proc. *IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2000, Vol. 1, pp. 472-479.

[6] G. B. Dantzig. *Linear Programming and Extensions*, Princeton NJ: Princeton University Press, 1963.

[7] C. Roos, T. Terlaky, and J.-Ph. Vial. *Theory and Algorithms for Linear Optimization: An Interior Point Approach*. Chichester: Wiley, 1997.

[8] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[9] T. Thiede, W. C. Treurniet, R. Bitto, C. Schmidmer, T. Sporer, J. G. Beerends, and C. Colones. “PEAQ - The ITU standard for objective measurement of perceived audio quality,” *J. Audio Eng. Soc.*, vol. 48, pp. 3-29, Jan.-Feb. 2000.