A FAST NOISE-SCALING ALGORITHM FOR UNIFORM QUANTIZATION IN AUDIO CODING SCHEMES

Carlos A. Serantes, Antonio S. Pena and Nuria G. Prelcic

E.T.S.E. Telecomunicación, Universidad de Vigo
36200 - Vigo (Spain)
apena@tsc.uvigo.es

ABSTRACT

A new bit assignment algorithm is presented. Its goals are the simultaneous assignment on all subbands in a few steps of an iterative calculus, the use of memory to achieve a better speed of convergence and the consideration of a deformable error curve. The basis of the algorithm is discussed and also other considerations that are likely to arise in practice. Finally, an example of performance is given.

1. INTRODUCTION

Simple bit assignment algorithms based on NMR (Noise to Masking Ratio) usually have the following characteristics:

- In each iteration, bits are given only to one band. Then, if we wish a precise control of the bit usage while using lossless compression, we are obliged to perform that operation many times, because assignment varies very slowly.
- The bit assignment chosen in previous frames, which is known by the coder, is not taken into consideration in the current frame.
- Only one error curve, independent of the bit rate, is used . So, if the bit rate is very low, quantization noise cannot be kept below the curve. Then we must choose between two options, either to keep the psychoacoustic profile or to modify it in order to obtain a perceptually optimal noise power distribution.

In some codecs, the side information associated with quantization consists of a group of scale factors and the number of bits assigned to each subband or set of transform coefficients. However, the ARCO (Adaptive Resolution COdec) coder [1] uses another approach. After computing the noise profile, obtained from a perceptual modeling and used to quantize a set of Wavelet Packet (WPK) coefficients, the curve must be sent to the receiver in order to recover the adaptive WPK tree structure [2]. To reduce the amount of side information, only this error curve and the quantization levels are sent.

The proposed algorithm [3] moves upwards and downwards the error curve till the bit usage is under the bit rate and within a certain range. The bit assignment is performed on all subbands at the same time, instead of only one band. The curve can be modified while searching for the correct shift, which can be estimated from the assignment carried out in the last coded frame.

2. BASIC STRUCTURE

The WPK coefficients are uniformly quantized using a predefined set of available quantizers so, to analytically justify this algorithm, the usual assumptions of zero-mean bounded random input are taken into account [4].

Under these conditions, when computing Δ_i (the quantizing step for band i) for a desired error variance Th_i, that can be scaled to satisfy the available number of bits for that signal excerpt, the following equation holds

$$\Delta_i = \sqrt{12 \cdot Th_i} \equiv \sqrt{12 \cdot Th_i \cdot scale} \equiv \sqrt{Th_i \cdot scale} \quad (1)$$

where the scaling value $scale$ is included and redefined to include the constant.

The minimum amount of bits to send M_i coefficients with L_i levels is, then, with K_i a function of both the estimated error power and the dynamic margin of the set of coefficients:

$$\begin{aligned} B_{infABS_i} &= M_i \cdot \log_2 [L_i] = M_i \cdot \log_2 \left[\frac{2 \cdot scf_i}{\Delta_i} \right] = \\ &= M_i \cdot \log_2 \left[\frac{2 \cdot scf_i}{\sqrt{scale \cdot Th_i}} \right] = \end{aligned}$$

$$= \quad M_i \cdot \log_2 \left[\frac{K_i}{\sqrt{scale}} \right] \qquad (2)$$

Obviously, first L_i, and after B_{infABS_i}, must be rounded to the next non negative integer, but let us consider the real number B_{infABS_i} as a lower bound to the minimum number of bits required to represent those M_i coefficients with L_i levels each.

The Equation 2 can be rewritten as a linear expression that relates B_{infABS_i} to the scaling value index n, if we force *scale* to be generated from an exponential expression

$$B_{infABS_i} \quad = \quad M_i \cdot \log_2 [K_i] - \frac{M_i}{2} \cdot n \cdot \log_2 [step]$$
$$if \quad scale = step^n, \quad n \in N \qquad (3)$$

So an approach to the final bit rate consumed by SB subbands, excluding the side information and the lossless compression processing, can be easily obtained from this property:

$$B_{infABS} = \sum_{i=1}^{SB} \left(M_i \cdot \log_2 [K_i] - \frac{M_i}{2} \cdot n \cdot \log_2 [step] \right) =$$
$$= \sum_{i=1}^{SB} M_i \cdot \log_2 [K_i] - \frac{n}{2} \cdot \log_2 [step] \cdot \sum_{i=1}^{SB} M_i \qquad (4)$$

Let us describe now the basic structure of the algorithm itself:

1. The total bit usage (B_{total}) is computed , including the lateral information, for $scale = 1$.
2. If B_{total} for that *scale* exceeds the available bit rate B, index n is increased.
    - The algorithm exits if B_{total} is within a fixed margin below B.
    - If B_{total} is above B, or below that margin, a slope to predict the scaling value that fits the desired bit rate is computed from two values for B_{total}, both first and current scaling values, taken into account the quasi-linear behaviour shown in Equation 4. This is iterated till the algorithm exits.

3. PRACTICAL CONSIDERATIONS

Figure 1 shows an example, where the bit usage is shown against the scaling value exponent, including the maximum amount of bits (1000) to use as a reference. For $n = 0$ the bit usage is 2030 bits and for $n = 1$, 1900. This provides a slope of 130 bits each n. So next try is $n = 8$ that estimates 990 bits, but the real value is found to be 1200 bits. So, reestimating the slope with the last two real values leads to 103.75 bits each n. Next try for n is then 10 to reach 993 bits, but as the real value is 1000 bits, the algorithm exits consuming only 4 iterations.

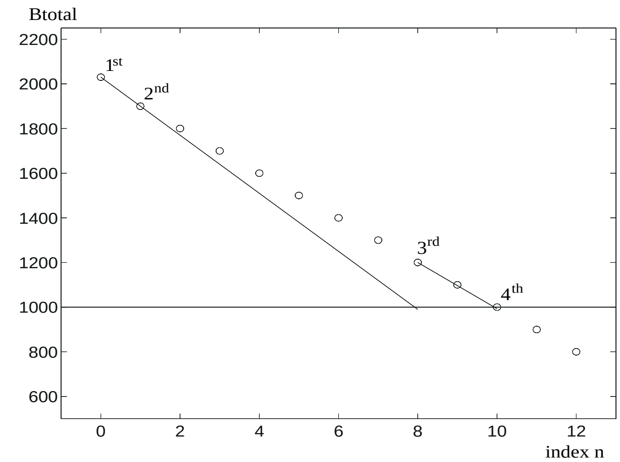

Figure 1: *Numerical example. Circles show the real bit usage for different scaling value exponents. Solid line follows the trajectory of convergence.*

There are several sources of non-linearities that must be considered in equation 4:

- Obviously, B_{infABS_i} cannot be negative so equation 3 is not linear but two-piece linear, as B_{infABS_i} becomes 0 for indexes n that hold

$$n > \frac{2 \cdot log_2 K_i}{log_2 step} \qquad (5)$$

  Then, equation 4 is also piece-wise linear with its slope changing every time a new subband is discarded in the bit assignment routine. One way to diminish this effect is to restrict the assignment to those subbands which are not likely to be zeroed. Another one is to avoid the use of very distant indexes when estimating the slope.

- The number of levels L_i is constrained to be an integer. Besides, its value is discrete, as the set of available quantizers is limited. Usually, only Midtread quantizers are considered to avoid the problem of representing zero entries. As a rule, the more quantizers we have, the better (except for the amount of side information).

- Sample grouping [5] improves linearity as each sample can be given a non integer amount of bits.

The use of this lossless compression technique is highly recommended.

- If other lossless compression routines are used, such as Huffman or Run Length Encoding (RLE), which are clearly non linear, the speed of convergence decreases. Nevertheless, the noise-scaling algorithm seems to perform well enough in practice.

Due to the existence of these non-linearities, a memory is necessary to keep the maximum known scale that makes $B_{total} > B$ and the minimum scale that makes $B_{total} < B$. Then, the scale given by the algorithm is constrained to be within the range defined by these bounds. If not, never ending loops could appear, as shown in figure 2.

There is a tradeback when choosing *step* in equation 4. If we make *step* larger, the algorithm will converge sooner, but to a rate that may not match well the available bit rate. On the contrary, if a smaller *step* is chosen, the algorithm will adjust the bit usage better, but may need more iterations to converge. Besides, the scaling index n should extend over a larger margin to be able to achieve the same scales as before. To sum up, the performance of the noise-scaling algorithm is highly adaptive.

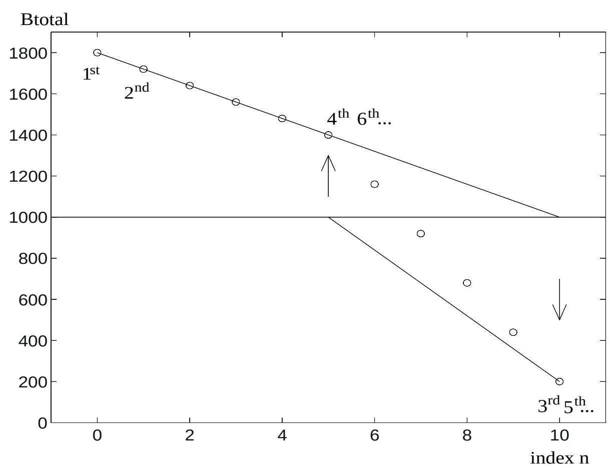

Figure 2: *Numerical example. A change of the linear behaviour results in an infinite loop.*

4. VARIATIONS

4.1. Prediction of the scale

Scale values usually vary slowly from frame to frame, even when the adaptive WPK tree structure has changed. Then, we can make an attempt with the scale used in the last frame to see if the equation $B_{total} \leq B$ still holds. If not, that scale is considered a good start point to search the correct one. In practice the number of iterations needed for convergence decays clearly. Of course, if the input signal is not stationary at all, the last scale may not resemble the current one, but it cannot be considered a bad choice to start either.

4.2. Variable scaling

Scaling values can be individually fixed for each band, that is, the noise-scaling algorithm allows to change the shape of the error curve, as in figure 3.

$$\Delta_i = \sqrt{Th_i \cdot scale_i} \tag{6}$$

This allows the algorithm to take into account some other considerations, p.e., noise loudness. Linearity shown by equation 4 is maintained because, in each subband, a fixed scaling is being used.

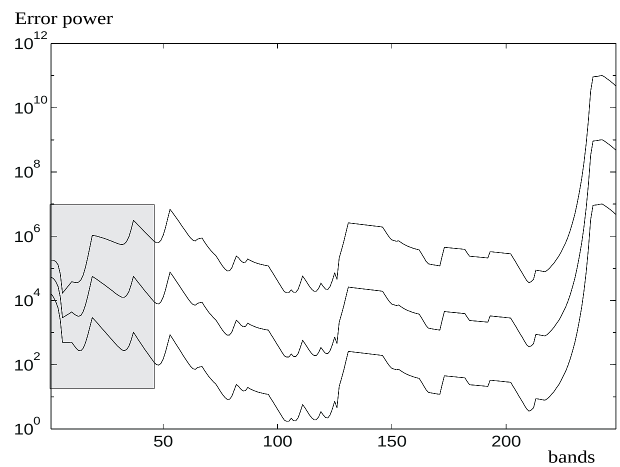

Figure 3: *Non uniform scaling of the shaded area.*

4.3. Overload error control

Using a limited set of quantizers may cause an overload error in some subbands, even for a scale that adjusts B_{total} to B very well. As power estimation is wrong under these conditions, it is advisable to make the scale larger till overload disappears, in spite of having a lot of unused bits at the end of the process. To avoid this problem, at least one of the quantizers should have a large number of levels, or some extra side information could be sent to the receiver.

4.4. Bandwidth variation control

If the bit rate is not high enough to provide a full band coded signal, a lowpass filtering may appear. The temporal change of the cut-off frequency from frame to frame may cause an on-off band switching that creates a very disturbing time pattern, commonly known as the *birdies* effect One way to solve it consists of making a minimum bit assignment within a certain range. If we wish to use the noise-scaling algorithm, a scalefactor structure is needed in some subbands. Obviously, if we want to avoid zeroing a subband for which Δ_i is very large, we must reduce either Th_i or $scale_i$. Besides, the receiver needs to know which bands have been given bits this way.

5. EXAMPLE OF PERFORMANCE

To evaluate the performance of the algorithm, a set of 32 quantizers was considered (Table 1).

Quantization levels
0, 1, 3, 5, 7, 9, 11, 13, 15
19, 23, 31, 39, 49, 63, 79, 101, 127 159, 201, 255, 321, 405, 511, 643, 811, 1023
1447, 2047, 2895, 4095
8191, 16383

Table 1: *Number of levels for each quantizer.*

These values were derived taking all the odd integers till 4 bits per sample, then 18 levels that are 1/3 bits away, 4 levels till 14 bits per sample (separated 1/2 bit), and a couple of large quantizers. The step was taken to be $2^{\frac{1}{8}}$ and RLE lossless compression was used both for samples and side information. Eight signals, containing stationary and non stationary excerpts, attacks, and all kinds of variations were taken into account.

Iterations	Scale indexes
6-3-2-2-2	158-154-154-154-154
7-4-5-3-3	190-193-197-196-195
8-3-2-3-2	179-177-177-178-178
10-5-5-6-7	151-140-109-133-143
5-4-3-4-4	149-157-158-160-170
8-5-4-4-2	159-184-173-167-167
11-4-3-2-4	143-142-143-143-149

Table 2: *Iterations and scale indexes obtained after using the noise-scaling algorithm with the parameters of the text.*

The results for the first five frames are shown in table 2. It is noticeable that for the first frame, when no memory can be used, the number of iterations is significantly larger. First and fourth examples are depicted in figure 4.

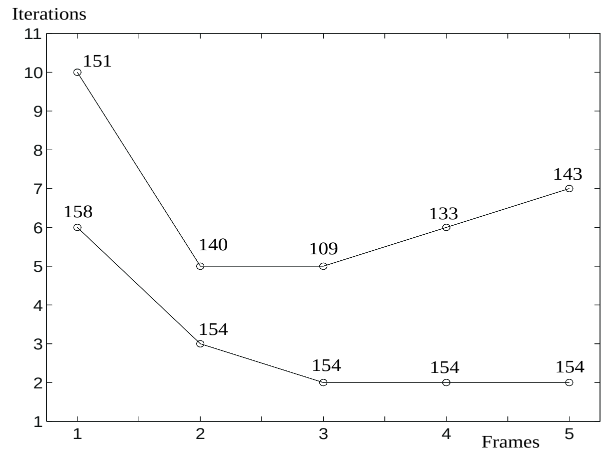

Figure 4: *Representation of two examples of performance.*

6. REFERENCES

[1] Antonio S. Pena, Carlos Serantes and Nuria G. Prelcic. *ARCO (Adaptive Resolution COdec): a Hybrid Approach to Perceptual Audio Coding.* AES 100th Convention 1996 May 11-14 Copenhagen.

[2] Nuria G. Prelcic and Antonio S. Pena. *An adaptive tree search algorithm with application to multiresolution-based perceptive audio coding.* 3rd IEEE Signal Proc. Soc. Int. Symp. on time-frequency and time-scale analysis, Paris 1996.

[3] Carlos A. Serantes. *Distribución eficiente de la información disponible en un codificador perceptual de audio basado en una transformada wavelet-packet adaptativa.* Master's thesis. E.T.S.I. de Telecomunicación. University of Vigo, Vigo 1996.

[4] N.S. Jayant and P. Noll *Digital coding of waveforms* Prentice-Hall, 1984.

[5] International Standard ISO/IEC IS 11172-3 *Information technology: Coding of moving pictures and associated audio for digital storage media up to 1.5 Mbps.- Section 3*, 1992