

# TRANSFORM CODING OF AUDIO SIGNALS AT 64 KBIT/S

Y. Mahieux and J. P. Petit

Centre National d'Etudes des Télécommunications  
Route de Trégastel 22300 Lannion FRANCE.

## ABSTRACT

The coding of high quality sound in 64 kbit/s is of high interest for applications such as ISDN. The algorithm that we describe in this paper, allows the reduction at such a bit rate while keeping the original quality. It is based on transform coding and it uses a TDAC transformation. Perceptual properties and the interblock redundancy of the spectrum are involved when coding the transform coefficients. The complexity of this algorithm allows its real time implementation on one floating point digital signal processor such as the DSP 32C of ATT. Finally, the performances and the subjective results of this coding system are discussed.

## I INTRODUCTION

The major drawback of the use of digital audio in Telecommunication applications such as ISDN, is the required bit rate. Many algorithms have been recently proposed to reduce this bit rate. Most of them are based on Transform Coding [1], [5]. They achieve a high quality coding at a bit rate of 2 to 3 bits per sample.

The coding system that we propose is designed to work at a bit rate of 64 kbit/s per monophonic channel for a sampling frequency,  $F_s$ , from 32 kHz to 48 kHz. But because of the flexibility of the bit allocation procedure, other bit rates are easily possible.

This algorithm looks like the well known Adaptive Transform Coding (ATC). Only the encoder is described in this paper. The structure of the decoder is quite symmetrical to the one of the encoder (figure 1).

After the TDAC transformation which is shortly described in section II, the masked coefficients are detected in the short term spectrum (section III). In order that encoder and decoder achieve the same bit allocation, a spectrum descriptor is computed from the non masked coefficients. Its coding scheme takes advantage of the long term stationarity of the signal (section IV). Then the section V describes the bit allocation procedure which is designed to perform the spectral noise shaping. The bits are distributed according to the perceptual importance of each coefficient. Finally, the section VI deals with the real time implementation and the performances of this algorithm.

## II TRANSFORMATION AND WINDOWING

The transform that we use is the Time Domain Aliasing Cancellation (TDAC) transform, [2], which is defined as :

$$y(m, k) = \sum_{n=0}^{N-1} x(n) \cdot h(n) \cos(2\pi(2k+1)(2n+1)/(4N) + (2k+1)\pi/4) \quad (1)$$

$k=0, \dots, N-1$

-N is the block size,  $h(n)$  is the analysis window;  
-m is the temporal index, k is the frequential index.

The inverse transform is :

$$x(m, n) = \sum_{k=0}^{N-1} y(m, k) \cdot f(n) \cos(2\pi(2k+1)(2n+1)/(4N) + (2k+1)\pi/4) \quad (2)$$

$n=0, \dots, N-1$ .

-f(n) is the synthesis window.

The reconstructed signal is simply obtained by overlap-add of  $x(m, n)$ . The perfect reconstruction is achieved with the windows  $h(n) = f(n)$  defined as :

$$h(n) = \sqrt{2} \cdot \sin(\pi \cdot (n+1/2)/N) \quad (3)$$

The computation of this transform is performed by a particularly efficient algorithm. The global amount of required operations is about  $N \cdot \log_2(N)$ . The coding algorithm has been designed with a  $N=1024$  sample window which delivers a relatively high spectral resolution. Because of the limitation of the human ear bandwidth, only the  $N_t$  first coefficients are coded.  $N_t$  corresponds to the frequency of 20 kHz. (see Table 1).

## III DETECTION OF THE MASKED COEFFICIENTS.

Because of frequential masking [3], some of the spectral lines  $y(m, k)$  are inaudible. They may be non transmitted without adding any perceptible degradation to the original signal. At the encoder, they are detected by comparing the energies  $y^2(m, k)$ , to the masking threshold  $S(m, k)$ . Then the masked coefficient indexes are transmitted to the decoder by "Run Length Coding".

405.2.1

00518

CH2827-4/90/0000-0518 \$1.00 © 1990 IEEE

Roku, Inc. EX1025

U.S. Patent No. 6,339,757

## 1) Computation of the masking threshold

### a) Principles of the computation

The computation of the masking threshold is performed on a line by line basis. First the spectrum is multiplied by the transmission factor of the ear,  $a_0(k)$  [3]. This weighting is equivalent to lower the energy of the highest frequencies. Then the masking threshold is derived from the convolution of the spectrum  $Y(\nu)$  expressed on the critical band rate (also called Bark scale),  $\nu$ , by the basilar membrane spreading function  $B(\nu)$ . The influence of each transform coefficient  $y(m,k)$  to the masking threshold of each other coefficient  $y(m,i)$ ,  $i \neq k$  can be expressed as :

$$S_k(m,i) = B(\nu_i - \nu_k) \cdot a_0(k) \cdot y^2(m,k) \quad (4)$$

The global masking threshold at the spectral line  $y(m,i)$  is :

$$S(m,i) = \theta \cdot \sum_{k=0}^{N_t-1} B(\nu_i - \nu_k) \cdot a_0(k) \cdot y^2(m,k) \quad (5)$$

The threshold  $\theta = 0.001$  corresponds to the signal to noise ratio for the complete masking of a tone by another tone located into the same critical band [3]. The basilar membrane spreading function  $B(\nu)$  and therefore the computation of equation (5) can be splitted into 3 parts. The first one, named central masking, refers to the values of  $\nu$  close to 0 where  $B(\nu)$  can be roughly approximated as a constant :

$$B(\nu) = 1 \quad -1/2 \text{ Bark} < \nu \leq 1/2 \text{ Bark} \quad (6)$$

The two other ones, called side masking, are the slopes of  $B(\nu)$ , that is the masking towards higher and lower critical bands (i.e. for  $|\nu| > 1/2 \text{ Bark}$ ). There,  $B(\nu)$  is estimated as a straight line whose slope is -10 dB per critical band towards higher frequencies and -27 dB per critical band towards lower frequencies (see the figure 2).

### b) Fast computation of $S(m,k)$

The TDAC transform delivers a regularly frequency spaced spectrum, whereas  $B(\nu)$  refers to the Bark scale which is a non linear function of the frequency. To solve this problem, the Bark scale is assumed to be constant over a low number of neighbouring coefficients.

On the other hand, a direct evaluation of the equation (5) would require a too high amount of computations. However, due to the peculiar shape of  $B(\nu)$ , it can be drastically reduced. For instance, for each coefficient  $y(m,k)$ , the side masking,  $S_{lb}(m,k)$  due to the lower frequency coefficients, can be partly derived from the previous one,  $S_{lb}(m,k-1)$ . In the same way,  $S_{ub}(m,k)$  (side masking resulting from higher frequency coefficients) can be computed from  $S_{ub}(m,k+1)$ . Actually, the 3 parts of the computation of  $S(m,k)$  are rewritten to a recursive form which is much more efficient regarding the computational load. Finally, the coefficient  $y(m,k)$ , is masked if :

$$a_0(k) \cdot y^2(m,k) < S(m,k) \quad (7)$$

The number of masked coefficients per transform block, is very variable. It is especially high for broadband signals. For instance, the average number of masked coefficients is 395 over 512 for a sequence of bassoon of duration 3 seconds.

## 2) Run Length Coding

### a) Coding principle

Let  $I(m,k)$  be a bit equal to 1 if the coefficient  $y(m,k)$  is masked. The series  $\{I(m,k), k=0, \dots, N_t/2-1\}$  exhibits more or less long unbroken runs of 0 or of 1. The length of each run is coded and transmitted, so allowing the decoder to rebuild the whole range  $\{I(m,k), k=0, \dots, N_t/2-1\}$ . Run Length Coding is efficient when associated with entropy coding [4]. Optimised Huffman codes have been computed in each case (runs of 0 and runs of 1).

The Run Length Coding starts after the first masked spectral line whose index is coded on 9 bits. At the decoder, the Huffman codes are decoded until the upper bound of the spectrum is reached. Then, the masked coefficients are simply forced to zero

### b) Run Length Coding restriction

To make sure that the non transmission of masked coefficients yields a positive gain of bits, the number of used bits,  $R_d(m)$  (i.e. the sum of the Huffman codeword lengths) is compared to the number of saved bits,  $R_s(m)$ .  $R_s(m)$  is the total number of the bits that would be allocated to the masked spectral lines if they were not forced to zero. For each transform coefficient this number is estimated by an average value previously computed. If the gain  $G(m) = R_s(m) - R_d(m)$  is negative, then this comparison is locally performed. At each run of masked coefficients, a local value  $G(m,k)$  is computed and the non transmission of the masked spectral lines is confined to the frequency regions where this gain is positive.

By using this procedure, the masking of coefficients is particularly attractive. The average ancillary bit rate is about 0.8 bit per masked coefficient.

## IV SPECTRUM DESCRIPTOR CODING

### 1) The Spectrum Descriptor computation

As shown in figure 1, a spectrum descriptor is transmitted at each block enabling the encoder and the decoder to compute the same bit allocation. For an efficient bit allocation procedure, this spectrum descriptor must be as precise as possible while requiring a low ancillary bit rate.

The spectrum descriptor is defined as the standard deviation  $\sigma(m,j)$  of the non masked coefficients in unequal width frequency bands. The number of bands,  $N_{sd}$ , depends on the sampling frequency (see Table 1).

## 2) Coding of the spectrum descriptor

The set  $\{\sigma(m,j)\}$  is largely uncorrelated. Coding techniques such as vector quantization are inefficient. But, the spectrum descriptor is highly time correlated. Indeed, the frequential distribution of the energy is slowly varying from one block to another. In case of transients, this temporal correlation vanishes. Therefore, the coding of the spectrum descriptor is performed by one coding scheme chosen among 2 different strategies.

### a) Temporal Predictive Coding

The most direct mean to take advantage of the inter-block redundancy, is to perform a predictive coding which is applied to the spectrum descriptor expressed in dB. This predictive coding scheme is made up of a convertor to log values, a DPCM encoder and a variable length encoding system (figure 3). The predictor is a first order one whose prediction coefficient  $a_1$  is very close to 1. A higher prediction order does not give a significant improvement.

The prediction error is :

$$e(m,j) = \log(\sigma(m,j)) - a_1 \log(\sigma'(m-1,j)) \quad (8)$$

$$j = 0, \dots, N_{sd} - 1.$$

–  $\sigma'(m,j)$  is the quantized version of  $\sigma(m,j)$ .

$e(m,j)$  is quantized by means of a 16 level uniform quantizer whose step size  $\Delta$  equals 5 dB. Because of the stationarity of the signal, the codeword density of probability is highly concentrated. By using an entropy coding of the quantizer outputs, less than 2.0 bits are sufficient instead of 4 bits.

### b) Frequential Predictive Coding

For this second quantization mode, a frequential predictive coding is performed. The prediction error is :

$$e(m,j) = \log(\sigma(m,j)) - \log(\sigma'(m,j-1)) \quad (9)$$

$$j = 1, \dots, N_{sd} - 1.$$

$e(m,j)$  is coded by means of a 16 level uniform quantizer also combined with a Huffman encoder. The first value,  $\sigma(m,0)$  is "log. quantized" by a 7 bit quantizer.

The spectrum descriptor coding mode is chosen according to the required amount of bits. The two predictive coding schemes are successively performed. The one which requires the lower bit rate is retained. To indicate this choice, one bit is transmitted to the decoder.

## V THE BIT ALLOCATION

### 1) General description

The bit allocation procedure distributes the bit rate between the spectral components, in such a way that the

quantization noise is masked. It is designed to minimize the ratio Coding Noise/Masking Threshold, i.e. for the block m

$$X(m) = \sum_{k=0}^{N_t/2-1} \sigma_q^2(m,k) / S(m,k) \quad (10)$$

–  $\sigma_q^2(m,k)$  is the variance of the quantization noise for the coefficient  $y(m,k)$ .  
–  $S(m,k)$  is the masking threshold.

$X(m)$  is minimized by including the constraint  $\sum R(m,k) = R_0(m)$ .  $R(m,k)$  is the number of bits allocated to  $y(m,k)$  and  $R_0(m)$  is the number of bits to distribute. This minimization yields the following equation :

$$R(m,k) = \frac{1}{2} \log_2(\alpha(m,k) \cdot \epsilon \cdot a_0(k) \cdot \sigma^2(m,k) / S(m,k)) + \lambda \quad (11)$$

where  $\lambda$  is a constant,  $a_0(k)$  is the transmission factor of the eqar.  $\epsilon$  is the performance factor of the quantizer [4].  $\sigma^2(m,k)$  is the variance of the coefficient  $y(m,k)$ . The additional factor  $\alpha(m,k)$ , allows to involve the flatness (or the peakness) of the spectrum [5].

At the decoder, only the quantized values of the spectrum descriptor are available. Then the number of bits is computed at each frequency group  $j$  from  $\{\sigma'(m,j)\}$ . All the non masked coefficients  $y(m,k)$  in the same band  $j$  receive the same number of bits. The masking threshold  $S(m,k)$ , previously computed for the detection of the masked coefficients is unavailable at the decoder. A new masking threshold  $S_1(m,j)$  is derived at the encoder and at the decoder from  $\{\sigma'(m,j)\}$ . Its computation is similar to the one described in section III-1.

$$\text{Therefore : } R(m,j) = 0.5 \cdot \Psi(m,j) + \lambda \quad (12)$$

where

$$\Psi(m,j) = \log_2(\alpha(m,j) \cdot \epsilon \cdot a_0(j) \cdot \sigma'^2(m,j) / S_1(m,j)) \quad (13)$$

$$j = 0, \dots, N_{sd} - 1.$$

### 2) Computation of $\Psi(m,j)$

The number of non zero coefficients in each frequency band is different (due to the non equal width of the bands and to the detection of masked coefficients). Then the performances of the quantizer are different too. Actually, 4 sets of quantizers have been computed (from 1 to 7 bits), each one corresponding to a different number of non zero coefficients in the frequency band.

The constant  $\alpha(m,j)$  is derived from the local spectral flatness measures ("sfm"),  $\{\gamma(\ell), \ell = 1, \dots, 4\}$ , which are computed in 4 frequency bands of equal width [4]. The 4 sfm's values are encoded by means of a 6 bit uniform quantizer whose range is [-64 dB, 0 dB]. The function  $\alpha(m,j)$  contributes to raise the perceptual importance of the peaks. If the spectrum is "sharper", the masking threshold has to be lowered [5]. So, for a better subjective quality, the peaks of the spectrum must receive a higher number of bits.

### 3) Bit Allocation procedure

By iterating the value of  $\lambda$ , the equation (12) is solved

so that  $\{R(m,j)\}$  are integer values which verify the constraint. The initial value of  $\lambda$  is given by :

$$\lambda_0(m) = (0.5 \cdot \sum_{k=0}^{N_t-1} \Psi(m,k) - R_0(m)) / (N_t/2 - N_{mc}) \quad (14)$$

$-N_{mc}$  is the total number of masked coefficients.

Then,  $R(m,j)$  is derived by

$$R(m,j) = I\{0.5 \cdot \Psi(m,j) + \lambda\} \quad (15)$$

where  $R_{\max} \geq R(m,k) \geq 0$  and  $I\{.\}$  means "nearest integer".

If  $\sum R(m,k)$  is close to  $R_0(m)$ , or if the number of iterations exceeds a certain value (e.g. 16), the procedure is stopped. Otherwise,  $\lambda_i(m)$  is modified :

$$\lambda_{i+1}(m) = \lambda_i(m) + (\sum R(m,k) - R_0(m)) / (F_{cs} \cdot N_t^i) \quad (16)$$

$N_t^i$  is the number of coefficients  $y(m,k)$  which received a positive number of bits at the step  $i$  and  $F_{cs}$  is the number of times that the sign of  $\sum R(m,k) - R_0(m)$  has been changing. So, when  $\sum R(m,k)$  is close to  $R_0(m)$ , the variation of  $\lambda_i(m)$  is gradually reduced. Then the procedure is reset to step 2.

Generally, less than 3 iterations are sufficient to reach the convergence state. If at the end of this iterating procedure,  $\sum R(m,k) \neq R_0(m)$ , bits are added or subtracted according to an algorithm similar to the one described in [6].

## VI IMPLEMENTATION AND RESULTS

### 1) Transmission error sensitivity

Although this algorithm is mainly devoted to ISDN applications with very low bit error rates, some precautions must be taken to avoid, for instance, the infinite propagation of errors. The sensitivity of the coding scheme to transmission errors is increased by the use of inter block prediction of the spectrum descriptor and of Huffman coding which generates a variable repartition of the bit rate inside the transform block.

To reduce this sensitivity, the bit stream may be rearranged and some ancillary informations have to be transmitted. For instance, the numbers of bits required by the Huffman codes (Run Length Coding and spectrum descriptor encoding) are transmitted at the beginning of each block. This allows the decoder to start the decoding of the remainder of the stream at the right place, even if a transmission error occurs.

Likewise, the temporal predictive coding of the bit stream is periodically reinitialised by using the frequential coding mode, even if the signal is stationary. So, the infinite propagation of transmission errors effects is avoided.

### 2) Complexity of the algorithm

The encoding algorithm requires about 12000 products and 18000 adds per block. Some particular mathematical operations have also to be performed (80 divisions per block and 120 logarithms). At the decoder, the number of operations is lower.

This computational load enables the real time implementation of the presented coding scheme. A floating point digital signal processor has been chosen in order to avoid the problems of precision which would be very critical with a fixed point format mainly because of the size of the transformation. The algorithm is now being implemented on a DSP 32C of ATT. At a sampling frequency of 32 kHz, one DSP 32C is sufficient to implement the encoder. The memory requirements are about 10 kBytes ROM and 12 kBytes RAM for the data aera.

This algorithm delivers a high quality sound at 64 kbits/s. For most pieces of music, it is very difficult to discriminate the original and the coded version. Subjective tests indicated that the algorithm works very well in case of broadband signals with many harmonics. For signals with flatter spectrum, the detection of masked coefficients is less interesting (or even does not work), yielding therefore a lower quality. This is particularly true for some critical pieces of song.

## CONCLUSION

Table 1 sums up the main characteristics of this coding algorithm and Table 2 presents a typical repartition of the bit rate inside one transform block. It is clear that the detection of masked coefficients and the predictive coding of the spectrum descriptor yields a high average number of available bits per non masked coefficient.

This algorithm is a good compromise between a high quality coding at 64 kbit/s and a limited complexity which allows its real time implementation. Finally, the coding scheme described in this paper is well suited for ISDN applications such as the transmission of a stereo high quality sound in a "2B" channel environment.

## REFERENCES

- [1] : K. Brandenburg & D. Seitzer, "Coding High quality audio with data rates of 64 kbit/s", 85<sup>th</sup> convention of AES; Nov. 1988.
- [2] : J. Princen, "Adaptive Transform Coding incorporating Time Domain Aliasing Cancellation"; Speech Comm.; Dec. 1987
- [3] : E. Zwicker, "Psychoacoustique"; Editions Masson 1981.
- [4] : N. Jayant & P. Noll, "Digital Coding of Waveforms"; 1984; Prentice Hall Signal Processing Series.
- [5] : J. Johnston, "Transform Coding of Audio Signals Using Perceptual Noise Criteria"; IEEE Journal on Selected Areas, Vol 6, No 2, February 1988; pp 314-323
- [6] : M. Perkins & T. Lookabaugh, "A Psychophysically Justified Bit Allocation Algorithm"; Proc. of ICASSP 1989; pp. 1815-1818

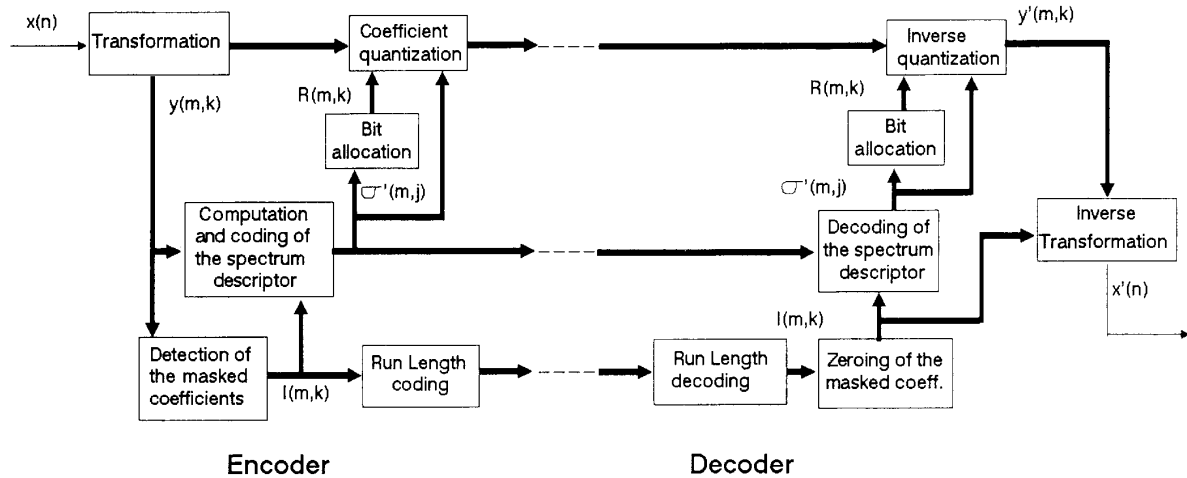


**Figure 1 General Scheme of the Algorithm**

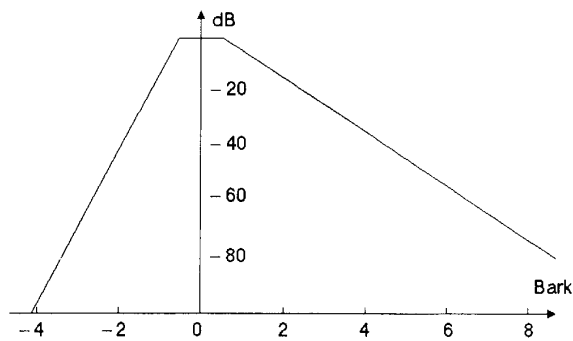


**Figure 2 Basilar membrane spreading function**

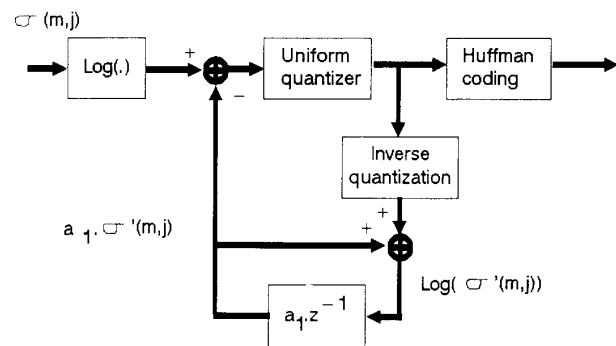


**Figure 3 Predictive coding of the spectrum descriptor**

|                                                                                           |                                  |
|-------------------------------------------------------------------------------------------|----------------------------------|
| Size of the Transform                                                                     | $N = 1024$                       |
| Maximal number of coded coefficients<br>sampling frequency $F = 32$ kHz<br>$F = 48$ kHz   | $N_t = N/2 = 512$<br>$N_t = 427$ |
| Number of available bits per block<br>$F = 32$ kHz<br>$F = 48$ kHz                        | 1024 bits<br>682 bits            |
| Number of frequency groups for the<br>spectrum descriptor<br>$F = 32$ kHz<br>$F = 48$ kHz | $N_{sd} = 50$<br>$N_{sd} = 35$   |

**Table 1 Main characteristics of the algorithm**

|                                                                      |                  |
|----------------------------------------------------------------------|------------------|
| Ancillary information<br>(Number of bits for Run Length Coding ....) | 21 bits          |
| The 4 local sfm's values                                             | 24 bits          |
| Number of masked coefficients                                        | 378 coefficients |
| Run Length Coding                                                    | 343 bits         |
| Coding of the spectrum descriptor                                    | 121 bits         |
| Bits to allocate to the non masked coefficients                      | 515 bits         |

**Table 2 Example of a bit stream repartition**  
(sampling frequency : 32 kHz).