Design and Implementation of a Nonuniform Subband Audio Coder

F. Argenti, A. Mannuccini, E. Del Re
Dipartimento di Elettronica e Telecomunicazioni – Università di Firenze
Via Santa Marta, 3 – 50139 Firenze
Italy
argenti@lenst.det.unifi.it – Ph. +39 055 4796 424 - Fax: +39 055 4796 485

Abstract

Subband coding is an efficient technique to compress audio and video signals. Signal spectrum is divided into subbands that are separately quantized and coded. In nonuniform filter banks, the subbands have different width. The use of nonuniform filter banks is attractive in some applications, such as audio coding, where the spectrum decomposition can be designed to match as close as possible the partition in *critical bands* of the human auditory system. This can simplify the subband quantization procedure, even if, as a drawback, the design and implementation of nonuniform filter banks is more troublesome than that of uniform ones. In this study, we describe a method to design nonuniform filter banks and the implementation of a real-time audio coder based on the TMS320C6201 DSP.

1. Introduction

In subband coding, the spectrum of a signal is divided into subbands, obtained by means of digital filtering and downsampling. Subband samples are quantized and coded. In the synthesis stage, the signal is reconstructed by means of upsampling and filtering. Techniques for the design of uniform filter banks, in which all filters have the same passband width, are well-established, whereas much less literature can be found for the nonuniform case [1]. The human auditory system can be represented as a bank of cochlear filters having nonuniform passbands (*critical bands*) [2], so that it can be modeled more easily in the nonuniform subband domain than in the uniform one. In coders based on uniform filter banks, such as MPEG-1 [3], a psichoacoustical model must run in parallel with the subband decomposition. The implementation of nonuniform filter banks, however, is less attractive than uniform ones, so that the pro and cons must be investigated.

In this work, a technique to design nonuniform filter bank is described. Several prototypes having different passband width are designed and modulated. Polyphase representation is used to exploit the advantages of modulation and downsampling in the analysis stage (upsampling in the synthesis stage). The nonuniform filter bank has been implemented on a TMS320C6201 Evaluation Module. The analysis and synthesis banks exploit efficient use of the pipeline. After time/frequency mapping, perceptual criteria are used to code the subband samples, assigning more quantization bits to subbands that are more sensitive to noise insertion. Unlike other audio coders, here the perceptual analysis is cascaded to the analysis bank, instead than computing a parallel psychoacoustical model.

2. Nonuniform filter bank design

An M-channel *nonuniform* filter bank with integer decimation factors is shown in Fig. 1. If all the downsampling/upsampling factors are equal to M, we have a *uniform* filter bank. In uniform M-channel cosine-modulated filter banks, the impulse responses of the analysis and synthesis filters are given by:

$$f_k(n) = 2\, h(n) \cos[\, (2k+1)\, (n-(N-1)/2) + \theta_k)\, \pi / (2M)\,]$$
$$g_k(n) = 2\, h(n) \cos[\, (2k+1)\, (n-(N-1)/2) - \theta_k)\, \pi / (2M)\,]$$

where $k = 0,1,\ldots,M-1$, $n = 0,1,\ldots,N-1$, and $\theta_k=(-1)^k\pi/4$. The N-length real coefficient prototype $h(n)$ is assumed symmetric. Its frequency response must satisfy, at least approximately, the power complementary (PC) property , i.e.,

$$|H(\omega)|^2 + |H(\pi/M- \omega)|^2 = M \qquad 0 \le \omega \le \pi/M$$
$$|H(\omega)|^2 + |H(-\pi/M- \omega)|^2 = M \qquad -\pi/M \le \omega \le 0$$

In nonuniform filter banks, several prototypes can be used [4-6], i.e., a different sequence $\{h(n)\}$ is used in each branch of the bank. Some prototypes, however, cannot be designed independently of the others [4]. Suppose two uniform M_1- and M_2-channel filter banks have been designed. Filters are selected from these banks to build the nonunifom one. Filters coming from different banks cannot have adjacent frequency responses. In fact, aliasing cancellation in the reconstructed signal does not occur since the prototypes for the M_1- and M_2-channel banks have, in general, different transition bands. An intermediate filter with transition bands matching the frequency responses of the neighboring filters is needed. This filter still has real coefficients, but it is obtained modulating a complex coefficient prototype with an asymmetric amplitude frequency response satisfying the PC property.
The nonuniform filter bank that has been used in this work to approximate the *critical bands* of the human auditory system has been designed by using this multi-prototype approach. Each prototype has been found with the eigenfilter method, an efficient and flexible technique to design a large variety of digital filters, both FIR and IIR [7-10]. In the eigenfilter approach, a weighted cost function E, measuring the approximation error between $H(\omega)$ and the desired target function, is expressed as a quadratic form, i.e., $E=y^T\mathbf{P}y$, where y is related to the filter coefficients and $\mathbf{P}$ is a positive-definite matrix [7]. The minimum of the quadratic form under the constraint $\|y\|_2 = constant$ is reached for y equal to the eigenvector corresponding to the minimum eigenvalue of $\mathbf{P}$. The eigenfilter approach can be applied to the design of linear phase prototypes, with real and complex coefficients satisfying the PC property, so that they can be used to implement uniform and nonuniform Near Perfect Reconstruction (NPR) modulated filter banks. The details are reported in [11]. Here we summarize the results inherent to the nonuniform filter bank implemented with the TMS320C6201 DSP for our audio coder.

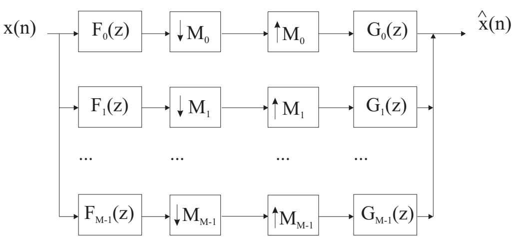

Figure 1. Nonuniform filter bank.

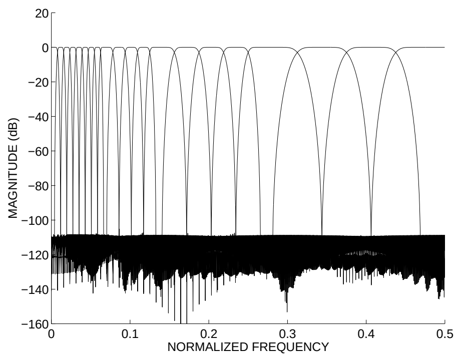

Figure 2. Analysis filters frequency responses of the nonuniform twenty-channel bank.

Table 1

P_i	N_i	M_i	NC_i
P_0	1024	64	8
P_1	512	32	3
P_2	256	16	3
P_3	128	8	3

Table 2

TF_i	N_i	M_i
TF_0	1024	32
TF_1	512	16
TF_2	256	8

Four real coefficient prototypes, denoted in the following as P_i, for i=1,…,4, have been designed for uniform filter banks with downsampling factors M_i equal to 8, 16, 32 and 64. NC_i adjacent filters have been selected from each uniform bank. These values, as well as the length of each prototype N_i, are summarized in Table 1. Three transition filters, denoted as TF_i, for i=1,2,3, have been designed as modulated versions of complex coefficient prototypes. They are used to fill the gaps left in the frequency domain between filters selected from different uniform banks. The length of each transition filter and the relative decimation factor are shown in Table 2. All the prototypes and transition filters have high stopband attenuation (greater than 110 dB). The final result is a twenty-channel filter bank, whose analysis filter frequency responses are shown in Fig. 2. The maximum global reconstruction error in the absence of aliasing is ±0.004 dB, whereas the maximum amplitude of the uncanceled aliasing components is -98.0 dB.

3. Implementation

In this section, the implementation of an audio coder based on the previously described nonuniform filter bank is presented. The filter bank, the psychoacoustical model and subband samples quantization are separately described.

3.1 Filter bank implementation

All passband filters - apart from the transition filters - are obtained by means of the cascade of prototype polyphase filtering and cosine-modulation. In an uniform cosine-modulated filter bank, the polyphase filtering shown in Fig. 3 can be used [12], where M is the decimation factor, the transfer functions $H_i(z)$ are the $2M$-polyphase components of $h(n)$, i.e., $h_i(n)=h(2Mn+i)$, and the matrix $\mathbf{C}$ contains the cosine terms, i.e., $[\mathbf{C}]_{m,n} = \cos[(2m+1)\ (n-(N-1)/2) + \theta_m)\ \pi / (2M)]$, $m=0,1,\ldots,2M-1$, $n=0,1,\ldots,(N/2M)-1$. The synthesis bank is implemented by using a different combination of these blocks. In a multi-prototype realization, the structure in Fig. 3 must be repeated several times. In our case, four $2M_i$-polyphase representations - one for each prototype P_i, $i=1,2,3,4$ - and four matrices $\mathbf{C}$ - containing, however, only the columns corresponding to selected filters - must be implemented.
Consider now the problems related to finite precision. The design procedure described in Section 2 provides floating point coefficients for the prototypes. Hence, the first operation is quantizing filter coefficients with a limited frequency response distortion. We used 16 bit words to store prototype coefficients. According to Table 1, the total number of coefficients to be stored for the real coefficient prototypes is 1920, i.e., 3840 bytes. They will be used in the analysis stage and, in reversed order, in the synthesis stage. The three transition filter coefficients are directly quantized with 16 bit words and stored. According to Table 2, the required memory is 3584 bytes.

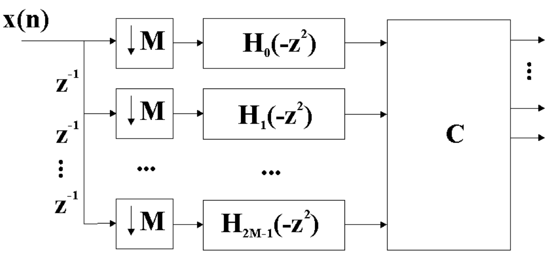

Figure 3. Polyphase implementation of a uniform filter bank.

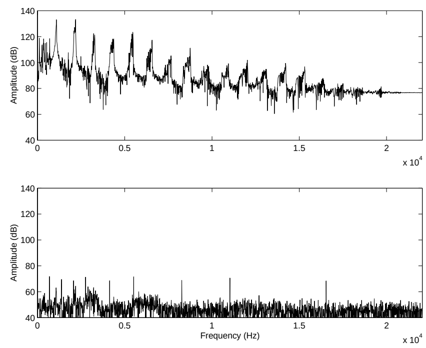

Figure 4. DFT of a 4096 samples segment read from an audio CD (top) and of the reconstruction error (bottom) obtained without subband samples quantization.

Cosine terms are quantized with 16 bit words and stored. For each prototype a different cosine terms matrix is used. The dimension of $\mathbf{C}$ are $2M_i \times NC_i$, i.e., they depend on the downsampling factor related to the prototype and on the number of channels produced with that prototype. Hence, according to Table 1, the total amount of memory for cosine terms is 2720 bytes. The synthesis stage uses $\mathbf{C}^T$: since the matrix product routine, discussed below, needs consecutive storing of cosine terms, they have been stored also in a different order to implement $\mathbf{C}^T$.
Suitable scaling factors have been computed to avoid overflow. In the analysis bank, overflow may occur after polyphase filtering and after cosine modulation, i.e., at subbands output. Similar checks are needed in the synthesis stage. Transition filters TF_i are implemented with a M_i-polyphase structure. Also for these filters, suitable scaling factors have been computed to avoid overflow.
The core of the filter bank implementation consists of the routines for FIR filtering and dot product, used for polyphase filtering and cosine matrix products, respectively. They are implemented by using fast assembler routines [13], whose characteristics are the following. The FIR filtering routine optimizes both memory access by reading signal samples and filter coefficients on a 32 bit word basis, and pipeline utilization by using *loop unrolling*. In the synthesis stage, prototype coefficients are read in reverse order. This routine allows to reach an almost optimal DSP utilization (close to 8 operations/clock cycle). Cosine-modulation is obtained as a matrix multiplication between polyphase filtered samples and cosine terms matrix $\mathbf{C}$. This operation uses a fast assembler dot product routine that allows to obtain 8 operations/clock cycle. In Fig. 4, the spectrum of an audio segment and that of its reconstruction after running the analysis/synthesis bank without samples quantization is shown.

3.2 Psychoacoustical model

From the discussion above, it is apparent that the implementation of a nonuniform filter bank is more expensive than that of a uniform bank, for which only one prototype filtering must be implemented. On the other hand, an advantage in implementing the psychoacoustical model is expected for nonuniform filter banks. In fact, if the subband position and bandwidth resemble that of human critical bands, the energy computed in each subband may be directly used in the computation of the masking threshold, representing the level of noise that can be injected into each band without noticeable distortion. Hence, if the quantization noise in each subband would be below the masking threshold, then it should not be perceived.

Table 3

Bark Index	Critical Band Edges (Hz)	Offset (dB)
0	0-100	-15.5
1	100-200	-16.5
2	200-300	-18.1
3	300-400	-19.05
4	400-510	-19.17
5	510-630	-23.1
6	630-770	-24.05
7	770-920	-24.29
8	920-1080	-24.29
9	1080-1270	-28.33
10	1270-1480	-26.0
11	1480-1720	-25.24
12	1720-2000	-23.81
13	2000-2320	-22.86
14	2320-2700	-21.59
15	2700-3150	-18.1
16	3150-3700	-18.1
17	3700-4400	-17.5
18	4400-5300	-17.0
19	5300-6400	-16.5
20	6400-7700	-16.0
21	7700-9500	-15.5
22	9500-12000	-15.0
23	12000-15500	-14.5
24	15500-	-14.0

Table 4

Channel Index	Passband Edges (Hz)	Bark Index
0	0-345	0,1,2
1	345-689	3,4,5
2	689-1034	6,7,8
3	1034-1378	9,10
4	1378-1723	11
5	1723-2067	12
6	2067-2412	13
7	2412-2756	14
8	2756-3445	15
9	3445-4134	16
10	4134-4823	17
11	4823-5512	18
12	5512-6890	19
13	6890-8268	20
14	8268-9646	21
15	9646-11024	22
16	11024-13780	23
17	13780-16536	23
18	16536-19292	24
19	19292-22050	24

We have used the following model. The decomposition into critical bands found in [2] is reported in Table 3. The actual frequency domain decomposition given by the implemented filter bank is shown in Table 4. At low frequencies, one or more critical bands can be associated to each filter (see third column of Table 4). At higher frequencies, more than one filter contribute to a single critical band. This mapping allows to estimate energy in each critical band. For the first four channels, the energy gathered by each filter is uniformly distributed among the critical bands falling into that subband. Since the critical bands are poorly selective, energy in each critical band infuences hearing of adjacent bands. Therefore, energy spreading among adjacent subbands has been computed. To this purpose, the spreading function proposed in [14] has been used, as in [15]. Afterward, it is assumed that low-frequency and high-frequency subbands have tone-like and noise-like characteristics, respectively. Tonality index influences the Signal-to-Mask ratio: noise-like signals are more capable than tone-like signals to mask noise. The offset between signal energy and masking threshold depends on the bark index. These offsets are shown in Table 3 (third column) for each critical band. The offset values have been derived from [16] for bands falling into the interval 0-5kHz. For higher frequency bands, we used an offset that has been experimentally derived. A similar approach has been used in [17]. At the end, the masking threshold is compared with the absolute threshold of hearing in quiet: the values $absthr\omega$ tabulated in [2] have been used. The reference level of the absolute threshold taken in each critical band is the minimum of all $absthr\omega$ falling into that band. The model described until now works into the bark domain. However, we have to assign quantization bits into the subband domain. Hence, the masking threshold assigned to the first four channel is the minimum of that computed for the critical bands they contain.

3.3 Bit allocation

The masking threshold computed as described in subsection 3.2 allows to define a Signal-to-Mask Ratio (SMR_i) for the ith channel. If we assign j bit to the ith subband, then a Signal-to-Noise Ratio

(SNR_i) can be derived [2]. Hence, the Mask-to-Noise ratio is defined as $MNR_i = SNR_i - SMR_i$. These values are used in a bit allocation procedure similar as in [2]. Then, subband samples are scaled and quantized according to 16 possible values of scalefactors and to the number of bits assigned to each subband. Bit allocation, scalefactors as well as quantized subband samples are transmitted.

4. Experimental results

The coder we have described has been implemented in real time on a TMS320C6201 Evaluation Module. The DSP receives a packet of 1024 audio samples, 16 bit/sample. Data are compressed with a given target bit rate and code bits are output. In the reconstruction process the order of these stages are inverted. A C++ program manages the exchange of data between the Host Port Interface of the DSP and the host as well as reading audio from an audio CD and playing the reconstructed signal on the host speakers. Several informal subjective tests have been made by using different types of music. We have found that transparent audio coding is achieved at bit rates in the range of 96-128 kbit/s. In some cases, good results are also obtained at lower bit rates, down to 64 kbit/s.

Conclusions

In this work, the implementation of a real time audio coder based on nonuniform filter banks has been described. The filter bank has been designed by using the eigenfilter approach. The filtering stage reaches almost optimal DSP utilization. A psychoacoustical model is cascaded to the filter bank, so that bits can be assigned to each subband according to perceptual criteria. Several aspects of the proposed coder need further investigation, e.g., a refinement of resolution at low frequency and the introduction of variable frame length.

References

[1] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, Englewood Cliffs, NJ: Prentice-Hall, 1992.

[2] B. Scharf, “Critical bands”, in *Foundation of Modern Auditory Theory*, New York: Academic, 1970, pp. 159-202.

[3] ISO/IEC IS 11172:3, “Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s - Part 3: Audio”, 1993.

[4] P.P. Vaidyanathan, T.Q. Nguyen, "Eigenfilters: A New Approach to Least-Squares FIR Filter Design and Applications Including Nyquist Filters", *IEEE Trans. Circuits Syst.*, Vol. CAS-34, no. 1, pp. 11-23, Jan. 1987.

[5] T.Q. Nguyen, "The Design of Arbitrary FIR Digital Filters Using the Eigenfilter Method", *IEEE Trans. Signal Processing*, Vol. 41, no. 3, pp. 1128-1139, Nov. 1993.

[6] S.C. Pei, J.J. Shyu, "Design of 1-D and 2-D IIR Eigenfilters", *IEEE Trans. Signal Processing*, Vol. 42, no. 4, pp. 962-966, Apr. 1994.

[7] F. Argenti, E. Del Re, "Design of IIR eigenfilters in the frequency domain", *IEEE Trans. Signal Processing*, Vol. 46, no. 6, pp. 1694-1698, June 1998.

[8] J. Princen, "The Design of Nonuniform Modulated Filterbanks", *IEEE Trans. Signal Processing*, Vol. 43, no. 11, pp. 2550-2560, Nov. 1995.

[9] S. Wada, "Design of Nonuniform Division Multirate FIR Filter Banks", *IEEE Trans. Circuits Syst. II*, Vol. 42, no. 2, pp. 115-121, Feb. 1995.

[10] F. Argenti, B. Brogelli, E. Del Re, "Design of Pseudo-QMF banks with rational sampling factors using several prototype filters", *IEEE Trans. Signal Processing*, Vol. 46, no. 6, pp. 1709-1715, June 1998.

[11] F. Argenti, E. Del Re, “Eigenfilter Design of Real and Complex Coefficient QMF Prototypes”, to appear in *IEEE Trans. Circuits Syst. II*, Vol. 47, no. 8.

[12] R.D. Koilpillai, P.P. Vaidyanathan, “Cosine-modulated FIR filter banks satisfying perfect reconstruction”, *IEEE Trans. Signal Processing*, vol. 40, no. 4, pp. 770--783, Apr. 1992.

[13] Texas Instruments Incorporated, “TMS320C62x/C67x Programmer's Guide”, Feb. 1998.

[14] M.R. Schroeder, B.S.Atal, J.L.Hall, “Optimizing digital speech coders by exploiting masking properties of the human ear”, *Acoustical Society of America*, Dec. 1979, pp. 1647-1652.

[15] J.D. Johnston, “Transform coding of audio signals using perceptual noise criteria”, IEEE Journal on Selected Areas in Communications”, 1988, pp. 314-323.

[16] M.A. Krasner, “The critical band coder--Digital encoding of speech signals based on the perceptual requirements of the auditory system”, *Proc. ICASSP*, 1980, pp. 327-331.

[17] D. Sinha, A.H. Tewfik, “Low bit rate transparent audio compression using adapted wavelets”, IEEE Transaction on Signal Processing , 1993, pp. 3463-3479.