How were the MP3 filter coefficients produced? *

Tom H. Koornwinder, KdV Institute, UvA, thk@science.uva.nl

version of February 19, 2007

A Mathematica notebook accompanying this paper can be downloaded from
http://www.science.uva.nl/~thk/art/misc/.

The audio coding scheme *MPEG 1 Layer-3*, which was standardized in 1993 and which is popularly known as *MP3*, has had an enormous impact for facilitating electronic distribution of music over the internet. Although various other more efficient coding schemes have been developed and implemented since then, the mp3 file format is still the only format for compressed audio files which is available on all platforms.

A lot of mathematics is being used in the MP3 coding. Various popularizing papers have appeared on this. However, it is not easy to get access to the precise formulation of the coding scheme and the precise mathematics behind it. Reasons for this are, on the one hand that the algorithm was developed by a community of signal analysts and computer scientists who use a terminology not familiar to mathematicians, on the other hand that this was developed in a commercial world, where access is far from free.

The MPEG 1 audio coding scheme [1] has started with Layer-1 and was soon extended to Layer-2, and next Layer-3. The higher Layers brought further compression while keeping a reasonable sound quality, among others by implementing the so-called psycho-acoustical model together with a better frequency resolution. Already in Layer-1, and preserved in the higher Layers, there is a splitting of the input signal into 32 subbands of equal frequency width. Reconstruction from the signals in these 32 subbands to the full signal is not completely exact, but by a very clever subband encoding/decoding scheme it is almost exact. The most detailed mathematical description of this scheme can be found in lecture notes by Schniter [4], available online but not formally published. Another reference is Pan [2]. See also the Remark at the end of this note.

The subband encoding/decoding scheme uses a prototype low-pass filter given by a list of 513 coefficients $C[i]$ $(i = 0, 1, \ldots, 512)$ for the analysis filter (see [1, Annex C, Table C.1]), which are between -0.04 and 0.04 and which are given with 9 decimals. Similarly, for the synthesis filter there is a list (see [1, Annex B, Table B.3]) of 513 coefficients $D[i]$ $(i = 0, 1, \ldots, 512)$, for which it turns out that $D[i] = 32C[i]$. There is more redundancy in the two tables because it turns out that $C[i] = -C[512 - i]$ if 64 is not a divisor of i, and $C[i] = C[512 - i]$ otherwise. Furthermore, $C[0] = C[512] = 0$.

*This is a slightly adapted version of my contribution to *Liber Amicorum Piet Hemker*, CWI, Amsterdam, November 17, 2006.

Define $h_{l+32k} := \frac{1}{2}(-1)^{\lfloor k/2 \rfloor} C[l+32k]$ $(l = 0, 1, \ldots, 31,\ k = 0, 1, \ldots, 15)$. Then $h_i = h_{512-i}$ and the ListPlot of the h_i (see Figure 1) is smooth. The algorithm uses the non-smooth $C[i]$ instead of the h_i for efficiency, in order to minimize the number of multiplications.

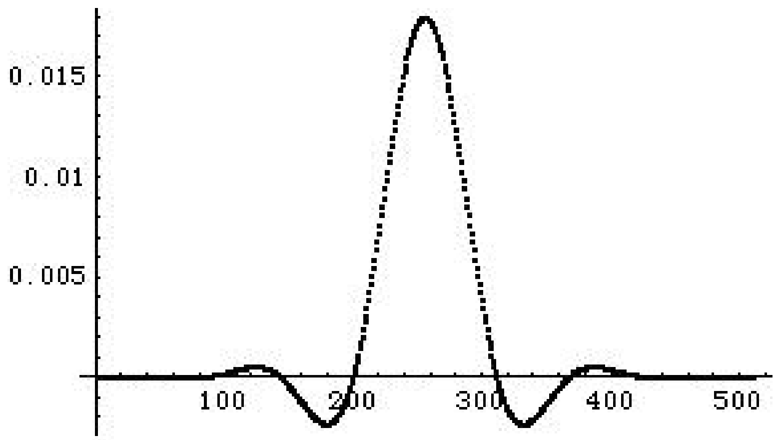

Figure 1: ListPlot for the filter coefficients h_i

The question I want to discuss in this note is how the coefficients h_i, in so many decimals, were obtained. Is there possibly a formula or algorithm yielding these coefficients? The official document [1] does not reveal the recipe being used[1]. However, see the Remark at the end of this note.

In order to get some idea about a possible answer to my question, let us first discuss the qualitative properties which the h_i should satisfy in order make possible the splitting of the signal into subbands with little frequency overlap, and allowing also almost exact reconstruction of the signal. Let $\phi(\xi)$ be the cosine polynomial which is essentially the Fourier transform of the h_i:

$$\phi(\xi) := h_{256} + 2 \sum_{n=1}^{255} h_{256-n} \cos(n\xi).$$

This is 2π-periodic and even, so we only need to draw the graph of ϕ on $[0, \pi]$, which we will consider separately on $[0, \pi/32]$ and on $[\pi/32, \pi]$, see Figures 2 and 3. The ideal Fourier transform is given by the block function in red in Figure 2: the function equal to 1 on $[-\pi/64, \pi/64]$ and equal to 0 outside this interval. This is not attainable with only finitely many filter coefficients. However, as shown by Figure 3, $\phi(\xi)$ almost vanishes for ξ outside the interval $[-\pi/32, \pi/32]$. This will cause that the subbands have some frequency overlap, but only if they are neighbouring. The clever subband encoding/decoding scheme is able to handle this during reconstruction.

[1]Schniter [4, p.16] writes: "Unfortunately, the standards do not describe how this filter was designed"

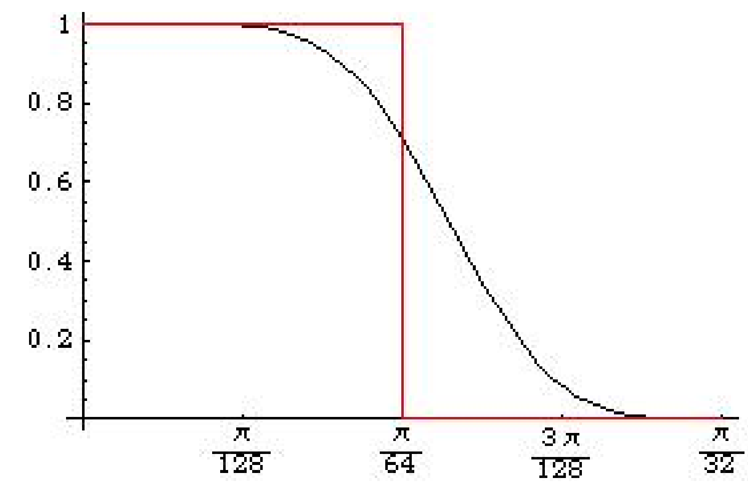

Figure 2: The Fourier transform $\phi(\xi)$ of the h_i on $[0, \pi/32]$

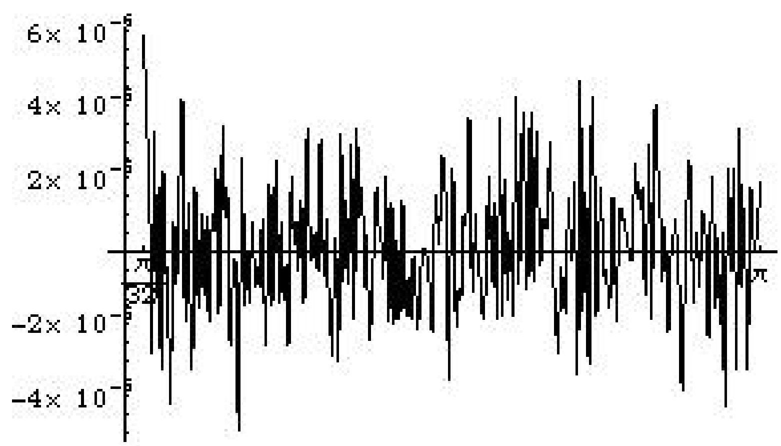

Figure 3: The Fourier transform $\phi(\xi)$ of the h_i on $[\pi/32, \pi]$

For optimal filter properties we should have the normalization

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\phi(\xi)^2\,d\xi = \sum_{n=1}^{511} h_n^2 = h\cdot h \approx \frac{1}{64}, \tag{1}$$

while

$$\frac{1}{2\pi}\int_{\pi/32}^{2\pi-\pi/32}\phi(\xi)^2\,d\xi = \frac{31}{32}\sum_{n=1}^{511} h_n^2 - \sum_{n\neq m} h_n h_m \frac{\sin((n-m)\pi/32)}{(n-m)\pi} = h\cdot h - Kh\cdot h \approx 0, \tag{2}$$

$$\text{where}\quad K_{mn} := \frac{\sin((n-m)\pi/32)}{(n-m)\pi}\quad \text{if } m\neq n,\quad \text{and } = \tfrac{1}{32} \text{ if } m = n.$$

The explicit h_n yield for (1) and (2), respectively:

$$h \cdot h - \frac{1}{64} = 7.2 \times 10^{-9}, \quad h \cdot h - Kh \cdot h = 2.9 \times 10^{-12}. \tag{3}$$

There is another condition necessary for almost perfect reconstruction:

$$\sum_{k=0}^{63} \phi(\xi + k\pi/32)^2 \approx 1 \quad \text{for all } \xi. \tag{4}$$

This implies (1). Furthermore, because of the symmetry and the almost vanishing of ϕ outside $[-\pi/32, \pi/32]$, (4) is equivalent to the condition

$$\phi(\xi)^2 + \phi(\pi/32 - \xi)^2 \approx 1 \quad \text{if } \xi \in [0, \pi/32]. \tag{5}$$

In terms of the h_n condition (4) is equivalent to

$$\sum_{m=32l+1}^{511-32l} h_{m-32l}\, h_{m+32l} \approx \tfrac{1}{64}\delta_{l,0} \quad (l = 0, 1, \ldots, 7), \tag{6}$$

where the case $l = 0$ is condition (1). For the explicit h_n and for $l = 1, 2, \ldots, 7$ the left-hand side of (6) respectively takes the values

$$6.7\times10^{-8}, \quad -1.8\times10^{-8}, \quad 7.6\times19^{-8}, \quad -9.6\times10^{-9}, \quad -6.4\times10^{-7}, \quad 2.9\times10^{-8}, \quad 1.9\times10^{-9}. \tag{7}$$

For a direct computation of the coefficients h_n satisfying conditions (2) and (6) I propose to start with (2). Let us try to solve this by looking for eigenvectors of the symmetric Toeplitz matrix K which have eigenvalue approximately 1 and which are even (i.e, invariant under reversion of the coordinates). Numerically this is not an easy exercise. A computation in Mathematica suggests that K has 9 eigenvalues equal to 1 up to 6 decimals and 487 eigenvalues equal to 0 up to 6 decimals. The other 15 eigenvalues are between 10^{-6} and $1 - 10^{-6}$. By a lucky accident there is an explicit symmetric tridiagonal matrix T commuting with K, given by

$$T_{j+1,j} = T_{j,j+1} = \tfrac{1}{2}j(511 - j), \quad T_{j,j} = (256 - j)^2 \cos(\pi/32), \tag{8}$$

see Slepian [5, §2.2]. Eigenvectors of K will also be eigenvectors of T. Slepian calls these eigenvectors *discrete prolate spheroidal sequences*. Mathematica readily produces an explicit orthormal system of eigenvectors of T which are either even or odd. This yields an othonormal system of 5 even eigenvectors of T, say $v^{(1)}, v^{(2)}, v^{(3)}, v^{(4)}, v^{(5)}$, which are eigenvectors of K with eigenvalue approximately 1.

Now we look for a linear combination $v = \sum_{j=1}^{5} c_j v^{(j)}$ which minimizes

$$\sum_{l=0}^{7} \left(\sum_{m=32l+1}^{511-32l} v_{m-32l}\, v_{m+32l} - \tfrac{1}{64}\delta_{l,0} \right)^2 .$$

We can find this in Mathematica by using the procedure `FindMinimum`, where we take as starting values $(c_1, c_2, c_3, c_4, c_5) = (1, -1, 1, 0, 0)$. The vector v thus obtained has marvellous properties, very similar to the properties of h. Figures 1 and 2 are exactly the same for v and its Fourier transform (which we will call ψ) as for h and ϕ. Figure 3 even becomes better for ψ than for ϕ: $|\psi(\xi)| \le 10^{-7}$ for $\xi \in [\pi/32, \pi]$. The values for (3) with h being replaced by v become

$$v \cdot v - \frac{1}{64} = -5.4 \times 10^{-11}, \quad v \cdot v - Kv \cdot v = 5.4 \times 10^{-14}.$$

For $l = 1, 2, \ldots, 7$ the left-hand side of (6) with h being replaced by v takes the respective values (compare (7)):

$$-7.0\times 10^{-11}, \quad -4.2\times 10^{-10}, \quad 1.2\times 19^{-8}, \quad -2.7\times 10^{-8}, \quad -9.1\times 10^{-7}, \quad 6.3\times 10^{-8}, \quad 1.4\times 10^{-9}.$$

The l^2 distance between h and v is 7.5×10^{-4}.

Altogether, it turns out that there is a rather quick way to find explicit filter coefficients in sufficient precision which are certainly as good as the filter coefficients given in [1]. But what is different for the h_n compared to the v_n is that the vector h, expanded in terms of the eigenbasis of K, has also nonvanishing terms with eigenvectors of K corresponding to eigenvalues significantly smaller than 1.

Let me conclude with a few observations which, for the moment, are still miracles for me:

1. Why are ϕ and ψ almost equal to 1 on $[-\pi/128, \pi/128]$?

2. Why can condition (4) or equivalently (5) be satisfied so well within the approximately 1 eigenspace of K, which is (restricting to the case of even eigenvectors) only 5-dimensional?

3. Why does the vector w with $w_{256} := h_{256}$ and

$$w_n := \frac{\sin((n-256)h_{256})}{\pi(n-256)} \exp(-h_{256}^2 (n-256)^2) \quad (n \ne 256)$$

give a very good approximation to h or v, however with Fourier transform behaving considerably worse than ϕ or ψ?

Remark After I finshed the earlier version (November 2006) of this note, I learnt that L. van de Kerkhof[2] (Philips, Eindhoven) has generated the filter coefficients as listed in [1]. He kindly wrote me that the standard in [1] was based on requirements described for instance in Rothweiler [3], and that an iterative gradient method was used for optimalizing the filter coefficients.

References

[1] *Information technology — Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s — Part 3: Audio*, ISO/IEC 11172-3:1993, International Organization for Standardization, 1993; `http://www.iso.org/iso/en/prods-services/popstds/mpeg.html`; `http://www.mp3-tech.org/programmer/docs/iso11172-3.zip`.

[2] see `http://en.wikipedia.org/wiki/MP3#Development`

[2] D. Pan, *A tutorial on MPEG/audio compression*, IEEE Multimedia 2 (1995), No. 2, 60–74; `http://das.iocon.com/res/docs/pdf/mpegaud.pdf`.

[3] J. Rothweiler, *Polyphase quadrature filters — A new subband coding technique*, in *Acoustics, Speech, and Signal Processing* (IEEE International Conference on ICASSP '83), IEEE, 1983, pp. 1280–1283.

[4] Ph. Schniter, *Sub-band coding*, EE-597 Class Notes, 1999; `http://www.ece.osu.edu/~schniter/ee597/`.

[5] D. Slepian, *Prolate spheroidal wave functions, Fourier analysis, and uncertainty — V: the discrete case*, Bell System Technical J. 57 (1978), 1371–1430.