A LOW DELAY FILTER BANK FOR AUDIO CODING WITH REDUCED PRE-ECHOES

Gerald Schuller
Institut für Theoretische Nachrichtentechnik und Informationstechnik
University of Hannover
30167 Hannover, Germany
email: schuller@tnt.uni-hannover.de

Abstract

A low delay filter bank for audio coding is presented. The low delay allows to reduce the length of pre-echoes generated by quantization errors. The structure provides perfect reconstruction and a system delay which can be pre-specified independent of the filter length. A robust algorithm for the design of such filter banks is developed. In an explained example for audio coding the length of the pre-echo is reduced to about one third for a filter bank with 128 equally spaced bands when compared to known filter structures.

1 Introduction

In audio coding filter banks are used for redundancy and irrelevance reduction. The usual requirements for such a filter bank is having filters with high stopband attenuation and little overlap between neighboring channels, so that the signal energy in one channel does not spread to neighbouring channels, which saves bitrate and makes the application of psychoacoustic models easier, and having many channels, which makes the redundancy and irrelevance reduction more efficient. Another important property is the system delay. It measures the time lag between the input signal of the analysis filter bank and the reconstructed signal at the output of the synthesis filter bank. Filter banks used so far in audio coding usually had system delays of $L-1$ samples, where L is the length of the filters used in the filter bank. Here the design of a low delay perfect reconstruction filter bank will be presented, where the system delay can be chosen independently of the filter length in the design process, i.e. the delay can be made smaller than $L-1$ samples. This is important for applications where a low system delay is required, like broadcast audio, where monitoring is important, but also for the reduction of so called pre-echoes in audio coding applications, as will be shown.

The source of the pre-echoes are quantization errors. Consider e.g. a signal consisting of only one impulse. The input of the synthesis filter bank will then consist of the downsampled and quantized impulse responses of the analysis filter bank. The quantization step size is adapted to the signal level of the analysis filter output by increasing it with an increasing signal level. The synthesis

filtering operation spreads the quantization noise over a time span of $2L - 1$ samples, determined by the length of the convolution of the analysis with the synthesis filter impulse responses. This leads to quantization noise which is preceeding the reconstructed impulse and is therefore called pre-echo. The length of the pre-echo corresponds to the system delay of the filter bank.

If the premasking level of the ear is exceeded by the pre-echo, the pre-echo becomes audible. Due to the temporal decay of the masking level the pre-echo is more likely to become audible with increasing length. The noise after the impulse is less critical because of the much slower decay of the postmasking level of the ear. Pre-echoes can occur not only before impulses, but the same is true for all large transients.

One approach to avoid pre-echoes is to keep the number of channels small, which keeps the filters and the system delay short. An example is the 32 band filter bank used in Layer 1 and 2 of the MPEG standard [13], which has a filter length of 512 taps and a system delay of 511 samples. But the low number of channels has the disadvantage of a limited potential for redundancy and irrelevance reduction.

Another approach is that of Layer 3. Its filter bank has 576 channels, with a filter length of 1632 taps and a system delay of 1631 samples. To avoid pre-echoes the number of channels can be switched to 192. This reduces the filter length to 864 taps and the system delay to 863 samples. This would still cause audible pre-echoes, therefore the quantization step size is also reduced if necessary, i.e. if the input signal consists of a impulse like signal. This has the drawback of an increased bit rate at the presence of these signals, and that a decision algorithm is needed for the switch between the different modes.

The filter banks used in the above coding schemes have symmetric impulse responses, so that the the quantization noise is spread equally before and after the impulse like signal. The low delay filter bank proposed here has asymmetric impulse responses, which spreads the quantization noise unequally. This can be viewed as a better match to the temporal masking properties of the ear.

Recently some new design methods for low delay filter banks emerged. Nayebi [6, 7, 8] showed that it is possible to design low delay filter banks. However, his design method produces no perfect reconstruction filter banks, and is not suitable for large filter banks because of the kind of optimization involved. The filter bank proposed here is described in more detail in [9, 10, 11]. The structure and its derivation is described in [9, 10], the 2 band case and the optimization in [11]. It is a modulated filter bank, which means its band filters are determined by a baseband prototype filter, one for the analysis and one for the synthesis, which simplifies the design and implementation. Here the structure of [9, 10] and the optimization of [11] is combined for the design of large filter banks with low system delay, and its performance in audio coding applications is shown.

2 The Low Delay Filter Bank

It has the following properties.

- Perfect reconstruction modulated filter bank
- Arbitrary filter length
- Realization with a fast algorithm
- Low system delay possible
- Filters with little overlap and high stopband attenuation possible
- Analysis can be different from synthesis filters

2.1 The Matrix Formulation

The polyphase representation is a convenient way to describe a filtering operation with subsequent downsampling [1]. Let

$$\mathbf{x}(m) = [x(mN), \ldots, x(mN + N - 1)]$$

be a vector of signal blocks of length N, where N is the downsampling rate and the number of channels (critical downsampling) and

$$\mathbf{y}(m) = [y_0(m), \ldots, y_{N-1}(m)]$$

be a vector of the N outputs of the analysis filter bank. Let $\mathbf{X}(z)$ and $\mathbf{Y}(z)$ be the z-transforms of $\mathbf{x}(m)$ and $\mathbf{y}(m)$ resp. The output of the analysis filter bank can then be written as

$$\mathbf{Y}(z) = \mathbf{X}(z) \cdot \mathbf{P_a}(z)$$

where $\mathbf{P_a}(z)$ is the polyphase matrix of the analysis filter bank,

$$\mathbf{P_a}(z) = \begin{bmatrix} P_{0,0}(z) & P_{0,1}(z) & \ldots & P_{0,N-1}(z) \\ P_{1,0}(z) & P_{1,1}(z) & & \\ \vdots & & \ddots & \\ P_{N-1,0}(z) & & & P_{N-1,N-1}(z) \end{bmatrix}$$

with

$$P_{n,k}(z) = \sum_{m=0}^{L-1} h_k(n + mN) z^{-m}$$

where L is the filter length in blocks of N samples. Here it can be seen that the synthesis for perfect reconstruction needs to be of the form

$$\mathbf{Y}(z) \cdot \mathbf{P_a}^{-1}(z) \cdot z^{-d} = \mathbf{X}(z) \cdot z^{-d}$$

where d determines the system delay. The multiplication with z^{-d} is needed in general to obtain causal synthesis filters. The total system delay n_0 is d blocks plus the blocking delay of $N - 1$

samples, $n_0 = d \cdot N + N - 1$. For practical filter banks $\mathbf{P_a}$ has to be constructed such that it has some important properties, like having an inverse which is FIR, having a structure which makes a fast implementation possible, or having a low system delay.

The approach taken here is to restrict $\mathbf{P_a}$ to be a product of different simple matrix types, all of which have FIR inverses, have an efficient way to compute the matrix product with them, different system delays associated with them, and result in a modulated filter bank. They are described in [9, 10, 11]. Some of them are presented below (the matrix elements not shown are zero).

- *Transform Matrix*– Denoted by $\mathbf{T}$. It can basically be any transform, but for the audio example a DCT type IV is used. Its elements are

$$t(n,k) = \cos(\frac{\pi}{N}(k+0.5)(n+0.5))$$

  $0 \leq n, k < N$

- *Coefficient Matrices* The first one has a diamond structure,

$$\mathbf{F} = \begin{bmatrix} 0 & & d_0 & d_N & & 0 \\ & \iddots & & & \ddots & \\ d_{N/2-1} & & & & & d_{N+N/2-1} \\ d_{N/2} & & & & & d_{N+N/2} \\ & \ddots & & & \iddots & \\ 0 & & d_{N-1} & d_{2N-1} & & 0 \end{bmatrix}$$

  The inverse has the same structure:

$$\mathbf{F}^{-1} = \begin{bmatrix} 0 & & d'_{N/2-1} & d'_{N/2} & & 0 \\ & \iddots & & & \ddots & \\ d'_0 & & & & & d'_{N-1} \\ d'_N & & & & & d_{2N-1} \\ & \ddots & & & \iddots & \\ 0 & & d_{N+N/2-1} & d_{N+N/2} & & 0 \end{bmatrix}$$

  The second one has a bi-diagonal structure,

$$\mathbf{C}_i = \begin{bmatrix} c_0^i & & & & & 1 \\ & \ddots & & & \iddots & \\ & & c_{N/2-1}^i & 1 & & \\ & & 1 & c_{N/2}^i & & \\ & \iddots & & & \ddots & \\ 1 & & & & & c_{N-1}^i \end{bmatrix}$$

Its inverse again has the same structure,

$$\mathbf{C}_i^{-1} = \begin{bmatrix} \hat{c}_0^i & & & & & & \hat{c}_N^i \\ & \ddots & & & & \iddots & \\ & & \hat{c}_{N/2-1}^i & \hat{c}_{N+N/2-1}^i & & & \\ & & \hat{c}_{N+N/2}^i & \hat{c}_{N/2}^i & & & \\ & \iddots & & & & \ddots & \\ \hat{c}_{2N-1}^i & & & & & & \hat{c}_{N-1}^i \end{bmatrix}$$

with real or complex coefficients.

- *Standard Delay Matrix*– It is cascaded in pairs with the coefficient matrices. It increases the filter length but also the system delay, because its causal inverse needs a multiplication by z^{-1}.

$$\mathbf{D}(z) = \begin{bmatrix} z^{-1} & & & & & \\ & \ddots & & & & \\ & & z^{-1} & & & 0 \\ 0 & & & 1 & & \\ & & & & \ddots & \\ & & & & & 1 \end{bmatrix}$$

Its causal inverse is

$$z^{-1} \cdot \mathbf{D}^{-1}(z) = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & 0 \\ 0 & & & z^{-1} & & \\ & & & & \ddots & \\ & & & & & z^{-1} \end{bmatrix}$$

- *Zero-Delay Matrices*– They also increase the filter length but they are not associated with any additional system delay because their inverse is causal. One example is

$$\mathbf{G}_i(z) = \begin{bmatrix} g_0^i z^{-1} & & & & & 1 \\ & \ddots & & & \iddots & \\ & & g_{N/2-1}^i z^{-1} & 1 & & \\ & & 1 & 0 & & \\ & \iddots & & & \ddots & \\ 1 & & & & & 0 \end{bmatrix}$$

Its inverse is

$$\mathbf{G}_i^{-1}(z) = \begin{bmatrix} 0 & & & & & 1 \\ & \ddots & & & \iddots & \\ & & 0 & 1 & & \\ & & 1 & -g^i_{N/2-1}z^{-1} & & \\ & \iddots & & & \ddots & \\ 1 & & & & & -g^i_0 z^{-1} \end{bmatrix}$$

With these matrices a polyphase matrix of a modulated filter bank can be constructed with the following cascade for the analysis filter bank.

$$\mathbf{P_a} = \left(\prod_{i=1}^{m} \mathbf{C}_i \cdot \mathbf{D}^2(z)\right) \cdot \mathbf{F} \cdot \mathbf{D}(z) \cdot \left(\prod_{i=1}^{n} \mathbf{G}_i(z)\right) \mathbf{T}$$

The synthesis polyphase matrix $\mathbf{P_s}$ (as in $\mathbf{Y}(z) \cdot \mathbf{P_s}(z) = \mathbf{X}(z)z^{-d}$) consists of the causal inverse term by term,

$$\mathbf{P_s} = \mathbf{T}^{-1} \left(\prod_{i=0}^{n-1} \mathbf{G}^{-1}_{n-i}(z)\right) \cdot \mathbf{D}^{-1}(z) \cdot z^{-1} \cdot \mathbf{F}^{-1} \cdot \left(\prod_{i=0}^{m-1} \mathbf{D}^{-2}(z) \cdot z^{-2} \cdot \mathbf{C}^{-1}_{m-i}\right)$$

The resulting length of the filters, the system delay, and the number of unknown coefficients of the filter bank is

$$\text{Length} = m2N + nN + 2N$$

$$\text{Delay} = m2N + 2N - 1$$

$$\text{Number of Coefficients} = mN + 2N + nN/2$$

There are more forms, like for minimum delay (see [9, 10, 11]). The Fast ELT [3, 4, 5] is a special case of this formulation. It is equivalent to the case of $n = 0$, $\mathbf{C}_i$ and $\mathbf{F}$ restricted to be orthogonal, and $\mathbf{T}$ to be a DCT type IV, which leads to orthogonal standard delay filter banks.

For the filtering operation, the number of multiplications necessary for the filter matrices ($\mathbf{F}$,C, $\mathbf{D}$, and G) is equal to the number of coefficients for the analysis filter bank (somewhat more for the synthesis filter bank), which is even less than the filter length. The transform can be computed with a fast DCT, which requires on the order of $N \log(N)$ multiplications. The small number of multiplications makes this filter bank suitable for the efficient implementation of large filter banks, which are usually needed for audio coding applications.

2.2 Optimization

The structure of the filter bank ensures perfect reconstruction and determines the system delay. Now the coefficients of the matrices have to be chosen in such a way that the frequency responses of the resulting filter bank comes close to some desired optimum. This is done by optimizing the matrix coefficients with respect to this desired optimum. There are many optimization methods, but methods that don't use information about the structure of the function to be optimized are usually slow or don't converge. The method described here briefly has the advantage of relatively robust and fast convergence for this application (see also [11]).

Define x to be a vector of the s unknown filter matrix entries, which are to be optimized, and $\mathbf{H}(\mathrm{x})$ be the weighted frequency response of the baseband prototype filter or one band filter at ℓ frequency samples. Analysis and synthesis frequency responses are both contained in this vector, e.g. as a concatenation, so that it has a length of 2ℓ. Let d be the vector of the weighted desired frequency responses at those frequency samples. As the error function a quadratic distance function was chosen. To optimize the magnitude of the frequency response, the following error function is used,

$$f(\mathrm{x}) = \sum_{i=1}^{2\ell} (|H_i(\mathrm{x})| - d_i)^2 = \sum_{i=1}^{2\ell} |H_i(\mathrm{x}) - \frac{H_i(\mathrm{x})}{|H_i(\mathrm{x})|} \cdot d_i|^2 =$$

$$\sum_{i=1}^{2\ell} |H_i(\mathrm{x}) - d_i'|^2$$

The minimization of this error function is done using the method of conjugate directions [12], tailored for this quadratic function. The idea is to use one-dimensional line minimization for finding the s dimensional minimum, and choosing the directions of the line minimizations carefully.

Line minimization can be done e.g. with Newtons method. To illustrate the idea, let x_0 be the starting point in the iteration, and v_i the unit vector in the direction of the line minimization. One Newton step per iteration is used,

$$\mathrm{x}_1 = \mathrm{x}_0 - \Delta\mathrm{x} \ , \quad \Delta\mathrm{x} = \frac{\partial f / \partial \mathrm{v}_i|_{\mathrm{x}_0}}{\partial^2 f / \partial \mathrm{v}_i^2|_{\mathrm{x}_0}} \cdot \mathrm{v}_i$$

The derivatives can be computed as

$$\frac{\partial f}{\partial \mathrm{v}_i} = 2\mathrm{Re}\{(\mathbf{H} - \mathrm{d}) \overline{\frac{\partial \mathbf{H}}{\partial \mathrm{v}_i}}^T \}$$

$$\frac{\partial^2 f}{\partial \mathrm{v}_i^2} \approx 2\mathrm{Re}\{\frac{\partial \mathbf{H}}{\partial \mathrm{v}_i} \cdot \overline{\frac{\partial \mathbf{H}}{\partial \mathrm{v}_i}}^T \}$$

where the overbar means complex conjugate. If $f(\mathrm{x}_1) > f(\mathrm{x}_0)$ then the magnitude of $\Delta\mathrm{x}$ is reduced, and if f is still bigger, x is left unchanged for this v_i. This ensures that the error function can only get smaller.

The s directions of the line minimizations, v_i, are determined by the eigenvectors of the Hessian matrix $\mathbf{B}$ of f. Usually it is computationally too expensive to compute the Hessian explicitly. But here the Hessian can be approximated with the first derivative of $\mathbf{H}$: $\mathbf{A} = \nabla \mathbf{H}^T$, where $a_{i,j} = \partial H_j / \partial x_i$.

$$\mathbf{B} \approx 2\mathrm{Re}\{\mathbf{A}\overline{\mathbf{A}}^T\}.$$

A new $\mathbf{B}$ is computed after the full previous set of eigenvectors v_i of $\mathbf{B}$ was used to update x. This is repeated until $|\Delta\mathrm{x}| < \epsilon$ for some $\epsilon > 0$. Note that only first derivatives of $\mathbf{H}$ are used for the optimization process and no stepsize parameter α is required.

This minimization process can be started with a random starting point. To make sure that a good minimum was found a second random starting point can be tried. For designing big filter banks (i.e. many bands, long filters), it can be faster to start with a smaller filter bank, i.e. to choose m and n to be small, like 0 or 1, and/or with a fraction of the desired numbers of bands. When the

optimization for this smaller filter bank is finished, m or n can be increased in steps of 2, with the coefficients of the added filter matrices initially set to 0, or the number of bands can be increased by increasing the size of the filter matrices, e.g. doubling the size and the number of bands by making pairs of coefficients out of each single coefficient. This is then the starting point for the optimization of the bigger filter bank. This process of growing the filter bank can be repeated until the desired size is reached.

3 Application to Audio Coding

This design method was used to design a filter bank for audio coding applications with 128 bands, a filter length of 1024 taps, and a system delay of only 255 samples ($n = 6,\ m = 0$, see Figures 2, 3, 4). It was designed such that the stopband attenuation has a falling slope, because this is advantageous to the application of psycho-acoustics. Observe the narrow bandwith and the attenuation in the neighboring channels (Figure 4). In Figure 2 it can be seen that the asymmetric impulse response of the baseband prototype matches the temporal masking function of the ear more closely than the symmetric impulse response of standard delay filter banks.

For a comparison a standard delay filter bank was designed with the same design method (Figure 5). It has the same number of channels, a similar frequency response, but a filter length of 768 taps and a system delay of 767 samples ($n = 0,\ m = 2$). This also shows that the system delay, and therefore the duration of a pre-echo, can be considerably reduced without reducing the filter quality. In this case the duration of a pre-echo for the low delay filter bank is only one third of the duration of the standard delay filter bank.

The comparison of the two is illustrated with castanets as an audio test signal, sampled at 44.1 kHz (Figure 6), which makes the system delay of the low delay filter bank 5.78 ms and the delay of the standard delay filter bank 17.39 ms. The quantization of the output of the analysis filter bank is simulated by adding noise with 10 dB SNR in each channel.

The result is that the filter bank with 768 samples delay produces audible pre-echoes before the hit of the castanets, whereas the filter bank with 255 samples delay features no audible pre-echoes. The low delay filter bank has filters which are not orthogonal. Indeed the SNR for its reconstructed signal is slightly lower than that of the standard delay filter bank, which has orthogonal filters, but the quantization noise of the low delay filter bank is not audible because it is masked by the signal.

4 Conclusion

A new filter bank design method is presented, which allows to design low delay perfect reconstruction filter banks. It is shown that it features significantly reduced pre-echoes compared to standard delay filter banks. The example with the 128 band filter banks shows that the low delay filter bank produces inaudible pre-echoes which have a duration of only about one third of those produced by a comparable standard delay filter bank, which are audible. This makes a higher number of bands and an improved frequency response possible, without the need to switch the number of bands. It leads to a more efficient redundancy and irrelevance reduction, and reduces the bit rate and the complexity of audio coders. The low number of multiplications necessary to compute the filtering operations makes it suitable for an efficient hardware implementation.

References

[1] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, 1993.

[2] A. Akansu and R. Haddad, *Multiresolution Signal Decomposition*, Academic Press, 1992.

[3] H. Malvar, *Signal Processing with Lapped Transforms*, Artech House, 1991.

[4] H.S. Malvar: "Extended Lapped Transforms: Fast Algorithms and Applications", ICASSP 1991, pp. 1797-1800

[5] H. S. Malvar: "Extended Lapped Transforms: Properties, Applications, and Fast Algorithms", IEEE Transactions on Signal Processing, Vol.40, NO.11, Nov. 1992.

[6] K. Nayebi, T. Barnwell, M. Smith, "Low Delay Coding of Speech and Audio Using Nonuniform Band Filter Banks," IEEE Workshop on Speech Coding for Telecom. Sept. 1991.

[7] K. Nayebi, T. Barnwell, M. Smith, "Design of Low Delay FIR Analysis-Synthesis Filter Bank Systems," Proc. Conf. on Info. Sci. and Sys., Mar. 1991.

[8] K. Nayebi, T.P. Barnwell,III, M.J.T. Smith: "Low Delay FIR Filter Banks: Design and Evaluation", Trans. on Signal Processing, January 1994, pp.24-31.

[9] G. Schuller and M. J. T. Smith, "A General Formulation for Modulated Perfect Reconstruction Filter Banks with Variable System Delay," NJIT 94 Sym. on Appl. of Subbands and Wavelets, Mar. 1994.

[10] G. Schuller and M. J. T. Smith,"Efficient Low Delay Filter Banks", DSP Workshop, Oct. 1994.

[11] G. Schuller and M. J. T. Smith, "A New Algorithm for Efficient Low Delay Filter Bank Design" ICASSP 95, Detroit, MI, May 1995.

[12] W.H.Press at al., *Numerical Recipes*, Cambridge University Press,1992

[13] ISO/IEC 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" International Standard, 1993

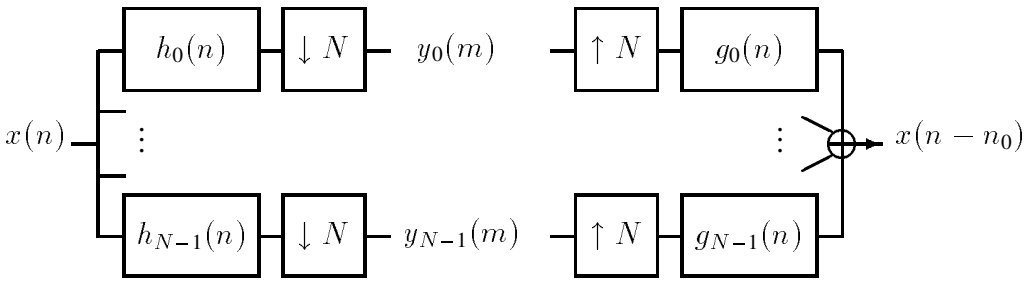

Figure 1: An N - channel filter bank with critical downsampling, perfect reconstruction, and a system delay of n_0 samples

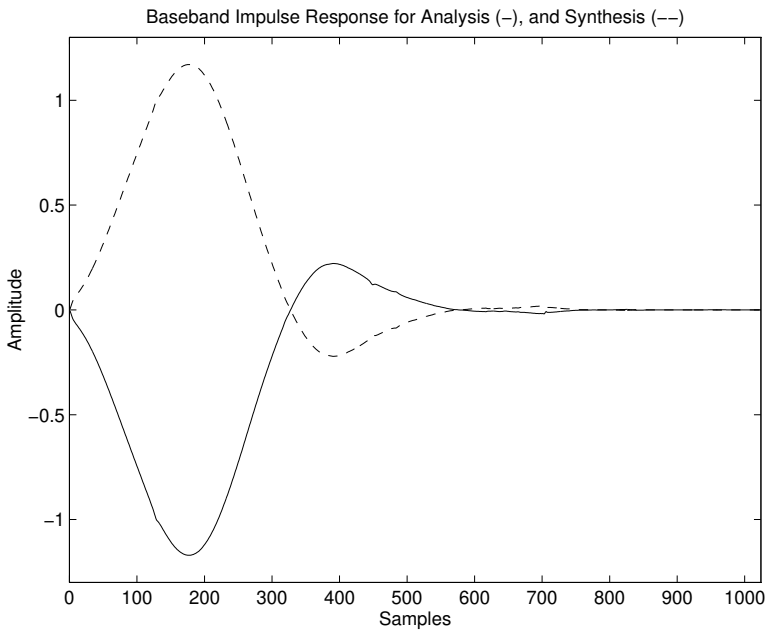

Figure 2: Impulse responses of the baseband prototype for the low delay filter bank, for analysis and synthesis.

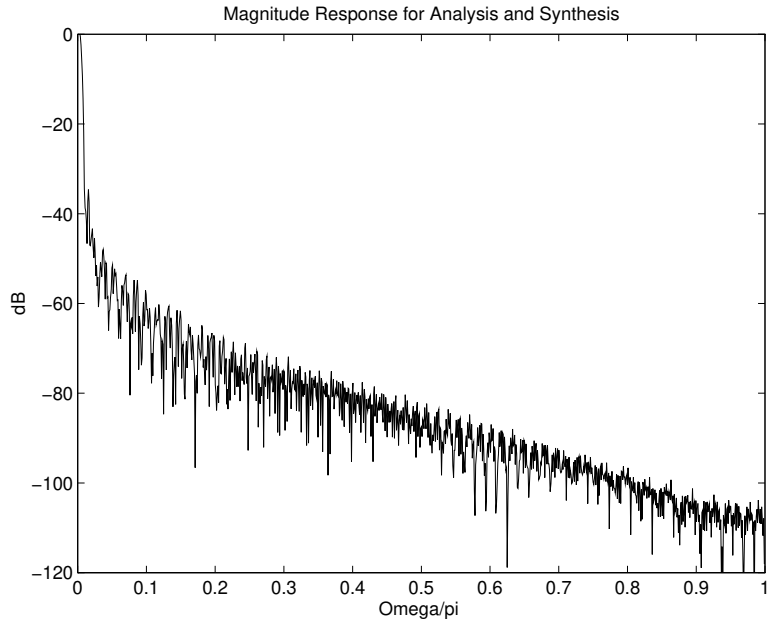

Figure 3: Magnitude responses of the baseband low delay prototype, identical for the analysis and synthesis filter bank.

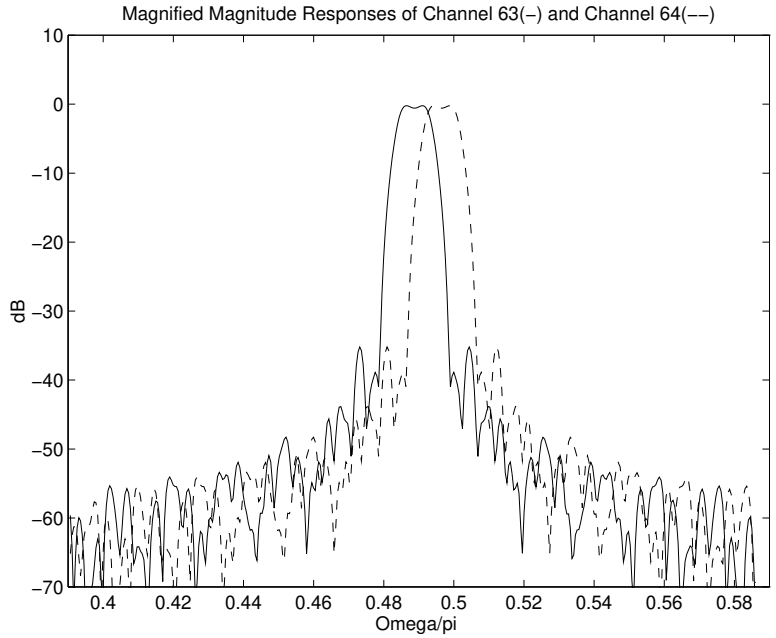

Figure 4: Close-up of the magnitude response of the low delay filter bank of channels 63 and 64 to show bandwidth and overlap between neighboring channels.

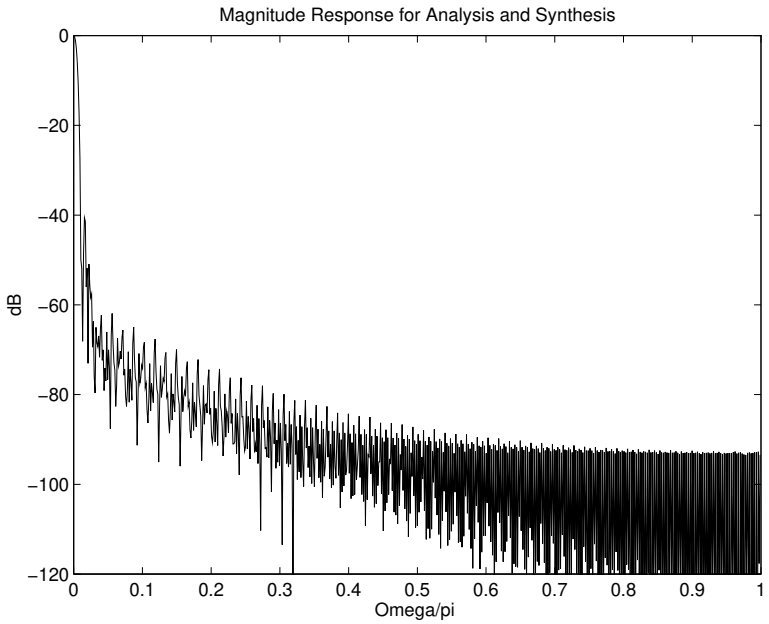

Figure 5: Magnitude responses of the baseband prototype for the standard delay filter bank, identical for the analysis and synthesis filter bank.

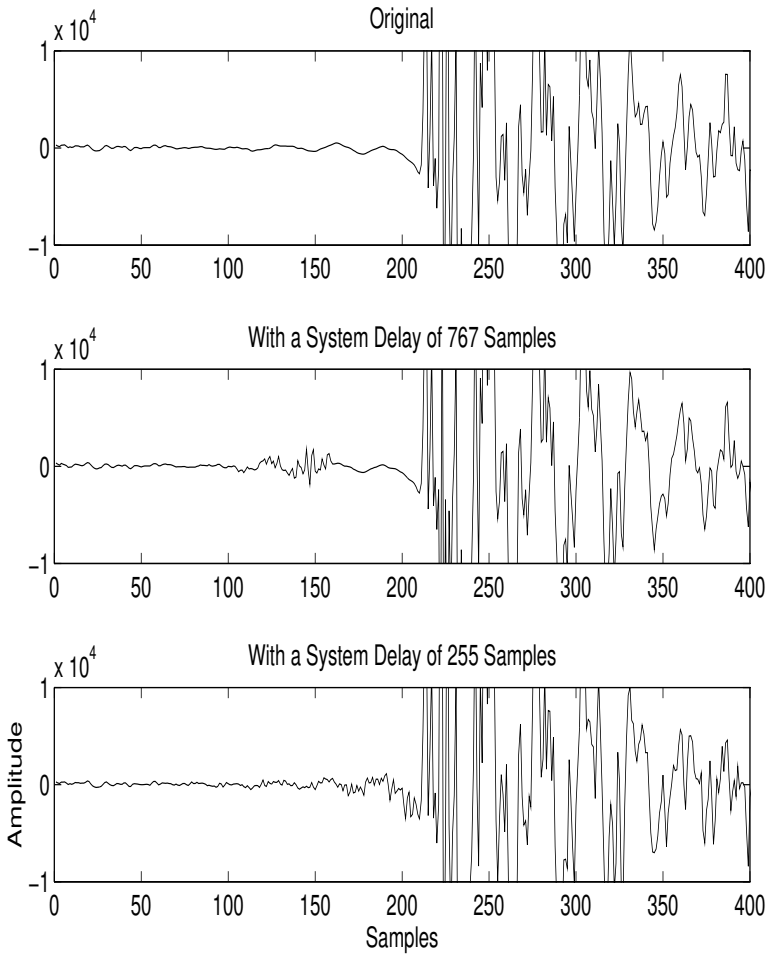

Figure 6: The system delay of 767 samples leads to an audible pre-echo, the system delay of 255 samples has no audible distortions. The pre-echo can be seen around sample 150.

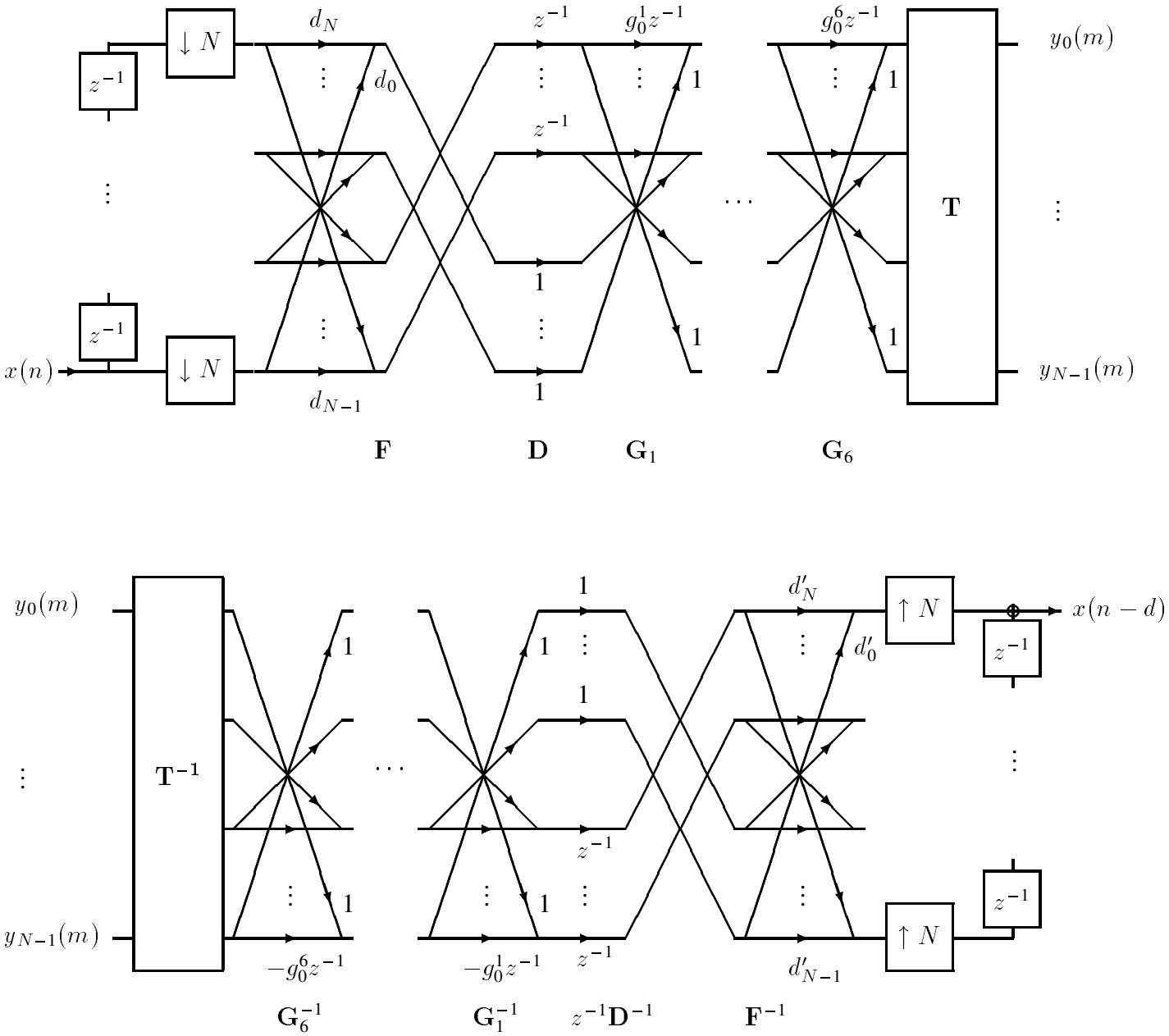

Figure 7: Structure of the low delay analysis filter bank (above) and the synthesis filter bank (below), with $m = 0$ and $n = 6$.