ARBITRARY FACTORISATION OF DECIMATION RATIO FOR EFFICIENT MULTISTAGE REALISATION

Vesa Lehtinen, Djordje Babic, and Markku Renfors

Tampere University of Technology, Institute of Communications Engineering
P.O.Box 553, FIN-33101 Tampere, Finland
vesa.lehtinen@tut.fi

ABSTRACT

Multistage implementation of decimators is in many cases more efficient than single-stage one. However, if the decimation ratio is prime, it cannot be factorised into a product of smaller integer factors. Previously it has been proposed that a rational decimation stage can be used to enable arbitrary selection of the decimation ratios. This way the advantages of multistage decimation and special filter structures such as halfband filters become available regardless of the overall decimation ratio. In this paper, we propose a method for optimising the overall response of such multistage decimators. Imaging and aliasing phenomena in such decimators are analysed, an iterative design method is introduced, and design examples are given. The proposed method can also be used for non-prime decimation ratios when their prime factors are not favourable for multistage implementation.

1. INTRODUCTION

Sample rate conversion (SRC) often requires less computation effort when implemented in multiple stages. Even though multistage implementation increases the computation rate of some filter coefficients, the overall computation rate and required coefficient storage are reduced. Moreover, multistage implementation relieves wordlength requirements of both signals and filter coefficients, which further reduces the computation effort. [1]

Another benefit of multistage SRC is that special filter structures can be used. For example, halfband filters are efficient in decimation by two because one of their polyphase branches is a pure delay [1]. Cascaded integrator-comb decimators [2] are efficient due to their simple, multiplier-free structure but they perform well only with relatively narrow stopbands. The benefits of both of these structures can be best exploited in multistage decimation.

Unfortunately, multistage implementation requires the decimation ratio to be factorisable. Thus, ordinary multistage implementation is not possible for prime conversion ratios. Even if the conversion ratio is not prime, its prime factors may not allow the full efficiency of multistage implementation.

Usually, disadvantageous SRC ratios can be avoided by system design decisions by choosing suitable sample rates. However, the coexistence of multiple different standards in, e.g., wireless communications and audio systems may lead to situations where there is little freedom to choose sample rates.

In [3], the authors propose the use of a stage decimating by a rational, noninteger ratio in order to allow multistage decimation when the overall decimation ratio is prime. The use of the rational stage enables arbitrary selection of decimation ratios. Usually, fractional SRC has been avoided when possible due to its (seemingly) higher complexity in terms of both computations and analysis of aliasing and imaging. The operation of efficient structures for fractional SRC is also somewhat harder to understand than that for integer SRC. However, the advantages of enabling efficient multistage implementation can compensate for and exceed the complexity penalty of a rational decimation stage.

In the following section, the proposed decimator structure is presented, and its imaging and aliasing phenomena are analysed. In Section 3, an iterative method for designing such decimators is introduced. Design examples are given in Section 4, and conclusions are drawn in Section 5.

2. THE DECIMATOR STRUCTURE

Multistage decimation and arbitrary factorisation of an integer decimation ratio L becomes possible if one stage is made to decimate by a rational ratio L/K. The product of the decimation ratios of the other stages must then equal K in order to obtain the overall decimation ratio L. Above, L and K are defined to be relatively prime, i.e., they do not have common factors. The proposed structure, and some equivalent structures used for analysis, are shown in Figure 2.1.

The parameter K is chosen so that it can be divided into factors K_n:

$$K = \prod_{n=1}^{S} K_n. \tag{2.1}$$

The rational stage and the integer stages can be placed in any order. The number of stages, S, as well as the values and order of the decimation ratios affect the efficiency of the system. The optimum also depends on the filter structures and implementation techniques used. Often it is best to place the rational decimation stage before integer stages in order to relieve its requirements [4].

The rational stage can be analysed using an equivalent structure shown in Figure 2.1(b) [5][6]. In this model,

This work has been supported by Nokia Foundation.

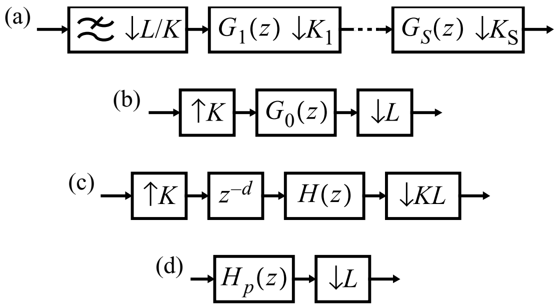

Figure 2.1. (a) The proposed multistage decimator. (b) An equivalent of the rational stage used for analysis. (c)(d) Structures equivalent to (a), used for analysis.

rational decimation by L/K is performed by upsampling by K, antialias/anti-image filtering and downsampling by L.

The multistage decimator can be analysed using the single-stage equivalent depicted in Figure 2.1(c). It can be observed that both the up- and downsampling ratios in this model are divisible by K. This leads to the equivalent structure of Figure 2.1(d). Here, $H_p(z)$ is one of the K polyphase components of $H(z)$ such that

$$h_k[n] = h[k + Kn], \quad k = 0, ..., K - 1 \tag{2.2}$$

and

$$H(z) = \sum_{k=0}^{K-1} z^{-k} H_k(z^K). \tag{2.3}$$

In this paper, $H(z)$ and $H_p(z)$ are referred to as the *unsampled overall transfer function* and *effective transfer function*, respectively. The value of p, i.e., which polyphase component appears as the effective response, depends on the delay d that appears together with $H(z)$ between the up- and downsampler in Figure 2.1(c). This delay models both the delays of actual filter stages and the possible phase difference between the input and output clocks of the rational stage.

The selection of the sampling phase p has an effect on both the magnitude and phase response. Even if $\{h[n]\}$ is symmetric, i.e., has a linear phase response, only one or two of its K polyphase components will be symmetric. It is even possible that there are no symmetric polyphase components at all. The number of symmetric polyphase components depends on K and the length of $\{h[n]\}$, the latter of which, in turn, depends on the stagewise orders and decimation ratios. Table 2.1 summarises the number of symmetric polyphase components.

Table 2.1: Number of symmetric polyphase components of $\{h[n]\}$.

		Length of $\{h[n]\}$	
		Even	Odd
K	Even	0	2
	Odd	1	1

Because the upsampling by K in the rational stage is cancelled by the integer stages, no images occur at the output of the last stage. All images of a given input frequency alias to the same output frequency, summing together coherently – all phase differences between them are due to the filtering

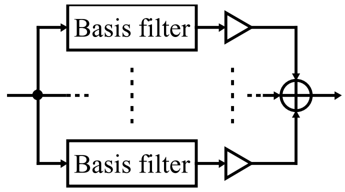

Figure 3.1. Decomposition of a filter (response) as a linear combination of basis filters.

and are independent of the signal.

If the overall decimation ratio is non-prime, the proposed method can be applied to any of its divisors. The remaining factors will then be implemented in ordinary integer decimation stages. Consider an overall decimation ratio of

$$L_{\text{tot}} = L \prod_{n=1}^{N_L} L_n, \tag{2.4}$$

where L will be factorised using the proposed method and L_n's will be implemented as integer decimation stages. The rational stage as well as the stages decimating by L_n's and K_n's can again be placed in any order.

3. FILTER OPTIMISATION

In this section, we introduce an iterative method for optimising the proposed class of multistage decimators in the minimax sense using linear programming.

3.1 Suboptimal design

Arbitrary factorisation can be applied using existing multistage filter design methods and tools. However, this means that the coherent summation of aliased images is ignored, which results in suboptimal performance. For example, even if the unsampled overall response is made equiripple, the effective magnitude response is not. The worst-case loss in stopband attenuation is $20 \log_{10} K$ dB.

3.2 Image-aware design

An equiripple magnitude response can be obtained by taking the aliasing of images into account in filter design. This can be equivalently done either in the frequency or time domain.The time-domain approach, based on the effective impulse response $\{h_p[n]\}$, is preferred due to its lower computational complexity. Thus, we obtain the following design problem: Minimise

$$\max_{\omega} W(\omega) \left| \left| H_p(e^{j\omega}) \right| - D(\omega) \right|, \tag{3.1}$$

where $D(\omega)$ and $W(\omega)$ are the desired magnitude and error weighting function, respectively.

Below, an iterative algorithm for optimisation of the proposed decimators is presented. Each filter stage is reoptimised multiple times, taking the responses of other stages into account.

As a tool for optimisation, each subfilter as well as the effective impulse response $\{h_p[n]\}$ can be decomposed into a linear combination of basis responses, illustrated in Figure 3.1. One basis response is needed for each independent coefficient of the filter under redesign. Such basis filters can be determined for many nonrecursive filter classes. Let $G_{s,k}(z)$'s form the basis of the s^{th} stage. When optimising $G_s(z)$, its basis filters are used for computing the basis transfer functions of $H_p(z)$, denoted as $H_{p,s,k}(z)$.

Here k is the index of coefficient. The basis transfer function $H_{p,s,k}(z)$ is obtained by substituting $G_{s,k}(z)$ for $G_s(z)$ and computing the effective response:

$$H_{p,s,k}(z) = H_p(z)\big|_{G_s(z) = G_{s,k}(z)}. \tag{3.2}$$

The basis transfer functions $G_{s,k}(z)$ depend on the filter structure. For example, for even-order symmetric direct-form FIR filters they can be written in the (non-causal) form

$$G_{s,k}(z) = \begin{cases} 1, & k = 0 \\ z^k + z^{-k}, & k = 1, 2, ... \end{cases} \tag{3.3}$$

The optimisation algorithm can be formulated as follows:

- Choose the filter parameters.
- Determine the sampling phase p. If two symmetric polyphase components of $\{h[n]\}$ are available, usually the longer one can be preferred since it can be expected to provide a slightly better performance.
- For each stage, design a starting point impulse response of the same length as that stage will have. Define as stopbands those frequencies that will alias to the passband or (lowest) transition band of the overall specifications. Choose filter parameters such that each stage reaches at least the stopband attenuation required from the overall response. The initial response does not necessarily need to be realisable by the chosen filter structure. The general shape of the frequency response suffices. For example, the Remez multiple exchange algorithm can be used for generating the initial responses.
- Starting from the last stage, redesign each stage in turn using the basis responses so that the effective response is optimised.
- Iterate the previous step multiple times. In the beginning, use a sparse frequency grid to maximise speed. Make it more dense for later iterations in order to obtain better precision. A few iterations should be enough if the initial responses are good.

If there are no symmetric polyphase components, linear programming can still be used in optimisation with little error if L is large. This is made possible by decomposing the asymmetric effective basis impulse responses into the sum of a symmetric and an antisymmetric component and ignoring the latter in optimisation; in other words, the average of the impulse response and its mirror image is used instead of the impulse response itself.

The choice of the initial stage responses is important. Experiments with Kronecker impulses as initial impulse responses led to bad performance and very slow convergence if any. Notice that the algorithm does not explicitly divide the filtering burden to the stages. In the optimisation of each stage, all frequencies of the overall specifications are optimised. This is quite different from ordinary iterative optimisation of multistage decimators. Instead, the division is implicitly determined by the initial responses – in a way, other stages form a weighting function for the stage being optimised.

Due to the coherence, the optimisation algorithm exploits destructive summation of aliased images. For this reason, the ripple levels of all stages will become approximately equal in minimax optimisation. Thus, it becomes difficult to find the bottleneck of the system by viewing the responses of the optimised filter stages, i.e., which stage should have its order increased. Instead, the bottleneck can often be found by comparing the ripple levels of the *initial* responses, at least if they are obtained from a filter of the same class as the actual filters used at that stage.

The rational stage can be implemented efficiently through polynomial interpolation using the transposed Farrow structure [7][8]. In such filters, the impulse response has a piecewise polynomial shape. Ultimately, if the overall decimation ratio is large, the rational stage can be implemented without computations as a hold-and-sample operation, i.e., the latest available input sample is used for each output sample. This is equivalent to a running sum FIR filter in place of $G_0(z)$ in Figure 2.1(b).

Table 4.1: Properties of example filters.

Design	Filter parameters	Mul. rate; fixed coefficients [multiplications / output sample]	Multiplication rate of basis multipliers of transposed Farrow [muls / output sample]	Weighted mul. rate [multiplications / output sample][a]	Delay [input sample intervals]
1-stage FIR	N=130	66	0	66[b]	65
1-stage tr. Farrow	N_0=14 M_0=4	35	76	63.5	132.5
2-stage	N_0=5 M_0=3 N_1=30	29	57	**50.4**	166
3-stage	N_0=5 M_0=2 N_1=14 N_2=38	53	38	67.3	225.5
4-stage	N_0=3 M_0=2 N_1 = 6 N_2=14 N_3=34	72	38	86.3	205

a. 6 bit multiplications required by basis multipliers were converted into equivalent 16 bit multiplications.
b. Here it is expected that the symmetry of the prototype filter is fully exploited; if not, the multiplication rate is 131.

4. DESIGN EXAMPLES

Three example multistage decimators were designed using the proposed method: two-, three-, and four-stage ones. As a reference for comparison, also two single-stage decimators were designed; one was a polyphase FIR filter, and the other was a transposed Farrow filter with bias-based symmetrisation [9]. The parameters and properties of the alternative designs are listed in Table 4.1. In the table, N_0 and M_0 stand for the number of polynomial segments and polynomial degree of the Farrow filter, respectively. The overall decimation ratio was 19, and the permitted ripple levels were 0.01 and 0.001 (–60 dB) for the passband and

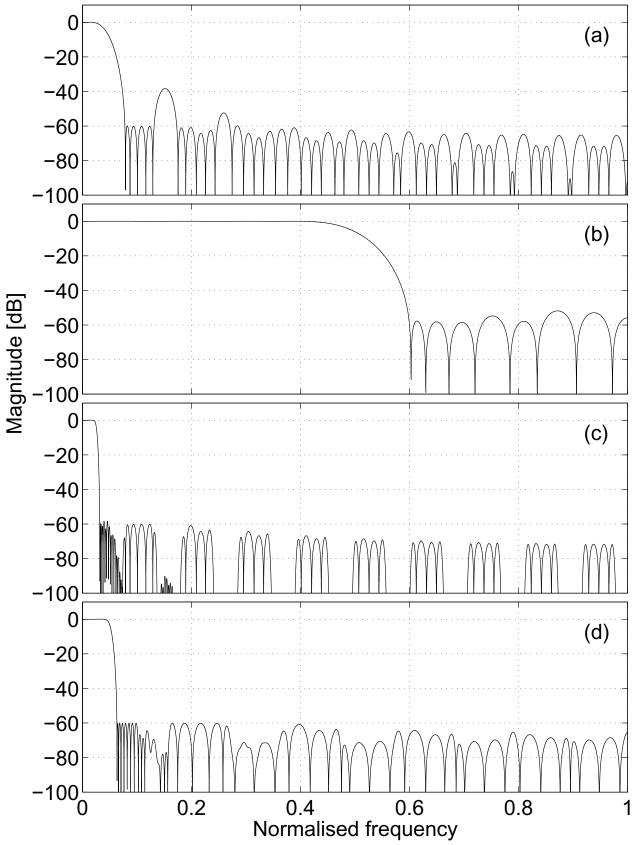

Figure 4.1. Magnitude responses of the two-stage design: (a) the rational stage, (b) the integer stage, (c) the unsampled overall response, (d) the effective response.

stopband, respectively. The passband and stopband edges were 0.4 and 0.6 times the output sample rate, respectively. In the multistage designs, the transposed Farrow structure was used at the rational stage, and halfband FIR filters were used at the integer stages, each thus decimating by two.

From Table 4.1 it can be seen that the two-stage design is the most efficient. It saves appr. 20% in the multiplication rate with respect to the polyphase FIR reference. Notice also that in the multistage structure it is easier to exploit the coefficient symmetry of FIR filters, especially when using halfband filters.

In Figure 4.1, the magnitude responses of the two-stage design are shown. It can be observed that the peak stopband ripple of the unsampled overall response exceeds the specified ripple limit slightly. However, these peaks are attenuated by destructive summation of aliased images, resulting in an adequate effective response.

Tightening the attenuation requirements would turn the situation more to the favour of the multistage designs, since larger wordlengths would be needed for fixed coefficients but the basis multipliers of the Farrow filter would remain unchanged.

5. CONCLUSIONS

In this paper, we have shown how rational decimation can be applied to enable multistage decimation or improve its efficiency. The usefulness of the proposed filter design method was demonstrated, and difficulties and pitfalls of the method were discussed.

The reference designs used in this paper were not the most efficient possible single-stage decimators. For example, decimators constructed from allpass polyphase branches [10] would probably be more efficient. If the integer stages are implemented with the most efficient algorithms, the efficiency of the rational stage will become the main bottleneck.

Unfortunately, the proposed method cannot be used for the optimisation of most recursive filter structures because they cannot be decomposed into a linear combination of basis filters. However, such stages can be optimised as if the upsampling by K were not present. Alternatively, the summation of aliased images can be taken into account in the frequency domain.

The proposed structure and optimisation method can be readily applied to interpolators as well.

6. REFERENCES

[1] R.E. Crochiere, L.R. Rabiner, *Multirate Digital Signal Processing*, Prentice-Hall, 1983.

[2] E.B. Hogenauer, “An Economical Class of Digital Filters for Decimation and Interpolation,” *IEEE Trans. Acoust., Speech, Signal Proc.* Vol. 29, No. 2, April 1981, pp. 155–162.

[3] H. Johansson, L. Wanhammar, “Two-Stage Polyphase Interpolators and Decimators for Sample Rate Conversions with Prime Numbers,” *Proc. VIII European Signal Proc. Conf.*, Trieste, Italy, Sept. 1996.

[4] T. Hentschel, G. Fettweis, “Sample Rate Conversion for Software Radio,” *IEEE Communications Magazine*, Aug 2000, pp. 142–150.

[5] W.H. Yim, “Distortion Analysis for Multiplierless Sampling Rate Conversion Using Linear Transfer Functions,” IEEE *Signal Proc. Letters*, Vol. 8, No. 5, May 2001, pp. 143–144.

[6] D. Babic, V. Lehtinen, M. Renfors, “Discrete-Time Modeling of Polynomial-Based Interpolation Filters in Rational Sampling Rate Conversion,” *Proc. IEEE Int. Symp. Circ. Syst.*, Bangkok, Thailand, May 2003, pp. 321–324.

[7] T. Hentschel, G. Fettweis, “Continuous-Time Digital Filters for Sample-Rate Conversion in Reconfigurable Radio Terminals,” *Proc. European Wireless*, Dresden, Germany, Sep 2000, pp. 55–59.

[8] D. Babic, J. Vesma, T. Saramäki, M. Renfors, “Implementation of the Transposed Farrow Structure,” *Proc. IEEE Int. Symp. Circ. Syst.*, May 2002, pp. IV-5–IV-8.

[9] V. Lehtinen, D. Babic, M. Renfors, “On Impulse Response Symmetry of Farrow Interpolators in Rational Sample Rate Conversion,” accepted to First International Symposium on Control, Communications and Signal Processing (ISCCSP), Hammamet, Tunisia, Mar 2004.

[10] M. Renfors, T. Saramäki, “Recursive Nth-band digital filters– Part I: Design and properties,” *IEEE Trans. Circ. and Syst.*, Vol. 34, No. 1, Jan 1987, pp. 24–39.