

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



---

# Two-Rate Based Structures for Computationally Efficient Wide-Band FIR Systems

Håkan Johansson and Oscar Gustafsson

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/52198>

---

## 1. Introduction

Many digital signal processing (DSP) systems tend to have a very high computational complexity when they target a large part of the Nyquist band. This corresponds to a wide-band system with one or several so called don't-care bands approaching zero. Examples of such systems include frequency selective filters, fractional-delay filters, and differentiators. This chapter considers finite-length impulse response (FIR) filters due to their attractive implementation features. In particular, they can be implemented with non-recursive structures. In contrast to infinite-length impulse response (IIR) filters, they are therefore always automatically stable and have no bound on the maximal sampling rate, see [1, 2].

For frequency-selective wide-band FIR filters, the frequency-response masking (FRM) technique can be employed for complexity reductions due to its use of sparse (namely periodic) subfilters, see [3–10]. For other functions, the FRM technique cannot be used directly, and one therefore has to seek other methods to reduce the complexity. This chapter discusses such a method which utilizes a two-rate technique, but only for the derivation of efficient single-rate structures. The basic two-rate approach was originally introduced in [11] and has since then been exploited and extended for various contexts as detailed in [12–19] and to be reviewed in this chapter. For single-function systems, it is however necessary to combine the two-rate technique with the FRM approach in order to achieve an overall complexity reduction. For multi-function realizations, complexity savings may be obtained without incorporating the FRM approach but it offers further complexity savings in such cases, as exemplified in [19]. Recent results have shown that the two-rate approach offers dramatic complexity reductions for wide-band systems, especially when combined with the FRM approach.

## 1.1. Chapter outline

Following this introduction, Section 2 considers the two-rate based structure that is appropriate for so called left-band and right-band systems which have don't-care bands at the low-frequency and high-frequency regions, respectively. Section 3 discusses the extension to so called mid-band systems which have don't-care bands at both the low-frequency and high-frequency regions. In Section 4, multi-function system realizations are considered, whereas Section 5 gives more implementation details. Finally, Section 6 concludes the chapter.

## 2. Two-rate based structure for left-band and right-band systems

This section will first revisit FIR filters and their computational complexity. After that, the two-rate based structure for left-band and right-band systems will be discussed.

### 2.1. Complexity of FIR filters

Consider a causal FIR filter with an impulse response  $h(n)$ , transfer function

$$H(z) = \sum_{n=0}^{N_H} h(n)z^{-n}, \quad (1)$$

and frequency response

$$H(e^{j\omega}) = \sum_{n=0}^{N_H} h(n)e^{-j\omega n}. \quad (2)$$

The order of the system is  $N_H$  and the impulse response duration (length) is  $N_H + 1$ . A direct-form implementation of the filter, corresponding directly to the convolution

$$y(n) = \sum_{k=0}^{N_H} x(n-k)h(k), \quad (3)$$

where  $x(n)$  is the input and  $y(n)$  the output, requires  $N_H + 1$  multiplications and  $N_H$  additions to compute each output sample  $y(n)$ . In the case of a linear-phase frequency response,  $h(n)$  is symmetric or anti-symmetric which reduces the number of multiplications to roughly  $N_H/2$ <sup>1</sup>.

The filter order required is determined by the application and specification. For example, for frequency selective filters, the order is inversely proportional to the transition band

<sup>1</sup> An even-order symmetric (anti-symmetric) linear-phase filter requires  $N_H/2 + 1$  ( $N_H/2$ ) multiplications whereas an odd-order linear-phase filter requires  $(N_H + 1)/2$  multiplications.

(don't-care band)  $\Delta = \omega_s - \omega_c$ , where  $\omega_c$  and  $\omega_s$  denote the passband and stopband edges, respectively, see [20, 21]. Hence, when the don't-care band decreases towards zero, the order increases rapidly. Then, using a direct-form realization, the computational complexity may become intolerable as it follows the filter order. The same trend exists also for other functions that are not frequency selective filters, like differentiation and integration, as seen in [22].

## 2.2. Two-rate based structure

To reduce the complexity, we consider here a structure that is derived via a two-rate approach, seen in Fig. 1. This structure is efficient for left-band systems (like a differentiator) targeting the frequency region  $\omega \in [0, \omega_c]$ ,  $0 < \omega_c < \pi$ . The same structure can also be used for right-band systems targeting the band  $\omega \in [\omega_c, \pi]$ ,  $0 < \omega_c < \pi$ . The only difference will appear in the design, and we will therefore focus on the left-band case in this chapter, and only comment upon the right-band case in the design section.

For a left-band specification, the basic idea is to first interpolate the input signal  $x(n)$  by two through upsampling by two followed by a lowpass filter with transfer function  $F(z)$ <sup>2</sup>. Then, a subsequent filter with transfer function  $G(z)$  follows that performs the actual function. Finally, downsampling by two takes place to retain the original sampling rate. Using multi-rate theory, see [23], it is readily shown that this scheme corresponds to a linear and time-invariant (LTI) system with a transfer function  $H(z)$  that equals the 0th polyphase component of the cascaded filter  $F(z)G(z)$ , i.e.,

$$H(z) = F_0(z)G_0(z) + z^{-1}F_1(z)G_1(z) \quad (4)$$

where

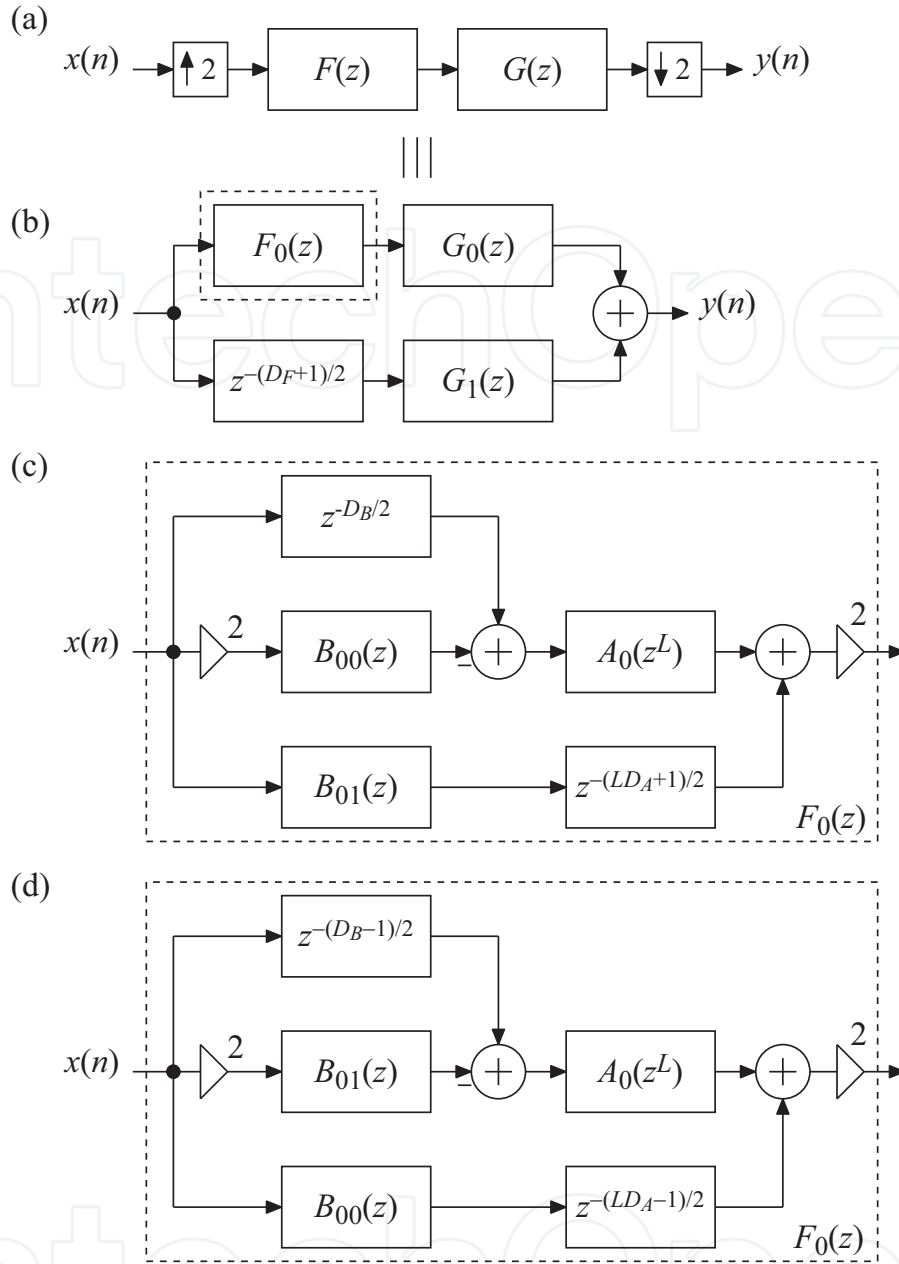
$$F(z) = F_0(z^2) + z^{-1}F_1(z^2) \quad (5)$$

and

$$G(z) = G_0(z^2) + z^{-1}G_1(z^2). \quad (6)$$

The final realization is thus a single-rate structure. A two-rate technique is only used to derive efficient structures. It is noted here that the order and delay of the overall filter  $H(z)$  is  $N_H = (N_F + N_G)/2$  and  $D_H = (D_F + D_G)/2$ , respectively. This can be understood by noting that  $F(z)$  and  $G(z)$  can be viewed as operating (in principle) at two times the input rate, because the structure is derived by sandwiching  $F(z)G(z)$  between upsampling and downsampling by two.

<sup>2</sup> The same function can be achieved by sampling the underlying analog signal with a higher sampling rate instead of sampling it slower and then use interpolation in the digital domain. However, this also increases the requirements on the analog-to-digital converters which are power-hungry components and in many cases one of the bottlenecks in overall systems. It is therefore often preferred to perform interpolation in the digital domain.



**Figure 1.** (a) Two-rate approach. (b) Equivalent LTI system when  $F(z)$  is an HB filter. (c) and (d) Polyphase component  $F_0(z)$  of the HB filter  $F(z) = F_0(z^2) + z^{-D_F}$  when realized with the FRM approach for an even-order (c) and odd-order (d) masking filter.

### 2.2.1. Filter Types

It is possible and efficient to let  $F(z)$  be a linear-phase half-band (HB) FIR filter<sup>3</sup>. Such a filter has a symmetric impulse response and every second impulse response value is zero, except the center tap which equals unity for an interpolation filter that preserves the signal energy. This corresponds to a pure-delay polyphase component  $F_1(z)$ , namely  $F_1(z) = z^{-(D_F-1)/2}$ ,

<sup>3</sup> If the delay is of importance, one may need to use a nonlinear-phase (approximately linear-phase) low-delay HB filter  $F(z)$  instead. Further, if there are additional requirements in the don't-care band, like attenuation requirements at  $\omega = \pi$ , a general filter  $F(z)$  must be used instead, i.e., a non-HB filter. See [19] for details.

Overall filter $H(z)$	Half-band filter $F(z)$	$G(z)$
Type I, even order $2(m + p + 1)$	Type I, order $4m + 2$	Type I, order $4p + 2$
Type II, odd order $2(m + p) + 1$	Type I, order $4m + 2$	Type I, order $4p$
Type III, even order $2(m + p + 1)$	Type I, order $4m + 2$	Type III, order $4p + 2$
Type IV, odd order $2(m + p) + 1$	Type I, order $4m + 2$	Type III, order $4p$

**Table 1.** Linear-phase filter types.

where  $D_F$  is the delay of  $F(z)$ , which is always an odd integer. When  $F(z)$  is a linear-phase filter,  $G(z)$  is of the same type as that of the overall filter  $H(z)$ , i.e., a linear-phase filter (nonlinear-phase filter) when  $H(z)$  is a linear-phase filter (nonlinear-phase filter). In this section, we focus on linear-phase filters. In the next section, nonlinear-phase applications are considered.

When  $F(z)$  is a linear-phase HB filter, it is a symmetric Type I filter with the odd-integer delay  $D_F$ . Its delay contribution,  $D_F/2$ , to the overall delay  $D_H$  is therefore an integer plus a half. It is consequently the delay contribution  $D_G/2$  of  $G(z)$  that determines whether the overall delay is an integer or an integer plus a half. As  $D_G/2$  then must be an integer or an integer plus a half to obtain an overall linear-phase filter,  $D_G$  must be an integer. Consequently,  $G(z)$  is either a Type I or Type III linear-phase FIR filter, i.e., an even-order filter with a symmetric or anti-symmetric impulse response. In other words, the type and order of  $G(z)$  determines the type and order of the overall filter, as summarized in Table 1. A formal proof of these facts is given in [17].

The order of  $G(z)$  is thus somewhat restricted as it cannot take on all even orders. However, the effective order of  $G(z)$  can be reduced by two by setting its first and last impulse response value to zero. In this way, two multiplications and additions may be saved in some cases. For the HB filter  $F(z)$ , it does not make sense to try to reduce the effective order in this way, as its impulse response is always zero for odd indexes of  $n$  (except for the center tap).

### 2.3. Complexity reduction

Assume that  $H(e^{j\omega})$  is to approximate a desired function  $D(j\omega)$  in the band  $\omega \in [0, \omega_c]$ . Due to the principle of interpolation by two in the two-rate based scheme, the effective bandwidth of  $G(z)$  is  $\omega_c/2$ , and thus always less than  $\pi/2$ . The complexity of  $G(z)$  alone will therefore be substantially lower than that of a regular direct-form realization of  $H(z)$ . (This will be discussed in more detail in the design example considered later in Section 2.4). However, the overall complexity is also determined by the filter  $F(z)$ . The requirement on this filter is roughly the same as that of the overall filter  $H(z)$  and its complexity is therefore relatively high. In other words, a major part of the overall complexity is moved to the filter  $F(z)$  and thus to  $F_0(z)$  in Fig. 1. Therefore, for a single-function system, there will not be any computational savings using this approach straightforwardly. This is because we can equally well combine the three subfilters into one single conventional filter.

Nevertheless, overall savings can indeed be obtained by utilizing additional complexity saving techniques for the lowpass frequency selective HB filter  $F(z)$ . Specifically, by realizing  $F(z)$  as an FRM filter, see [3–10], we can express the transfer function as

$$F(z) = 2A(z^L)B_0(z) + 2[z^{-LD_A} - A(z^L)]B_1(z) \quad (7)$$

where  $A(z^L)$  is a period model filter and  $[z^{-LD_A} - A(z^L)]$  is its complement, whereas  $B_0(z)$  and  $B_1(z)$  are masking filters. Specifically, in the case of a HB filter, as detailed in [5, 7],  $A(z)$  is given as

$$A(z) = A_0(z^2) + 0.5z^{-D_A}, \quad (8)$$

with  $D_A$  being the delay of  $A(z)$ , whereas the masking filter are related according to

$$B_1(z) = z^{-D_B} - (-1)^{D_B} B_0(-z), \quad (9)$$

with  $D_B$  being the delay of  $B_0(z)$ . One then finds that  $F_0(z)$  becomes, for  $D_B$  even:

$$F_0(z) = 2z^{-(LD_A+1)/2} B_{01}(z) + 2[2B_{00}(z) - z^{-D_B/2}] A_0(z^L) \quad (10)$$

and, for  $D_B$  odd:

$$F_0(z) = 2z^{-(LD_A-1)/2} B_{00}(z) + 2[2B_{01}(z) - z^{-(D_B-1)/2}] A_0(z^L) \quad (11)$$

where  $B_{00}(z)$  and  $B_{01}(z)$  are the polyphase components of  $B_0(z)$ , i.e.,  $B_0(z) = B_{00}(z^2) + z^{-1}B_{01}(z^2)$ . The resulting structures for  $F_0(z)$  are depicted in Fig. 1(c) and (d). More details can be found in [7].

As seen,  $F_0(z)$  makes use of three subfilters, of which  $A_0(z^L)$  is periodic for an integer  $L > 1$ . A periodic filter is a sparse filter, meaning it has many zero-valued filter coefficients. Specifically, only every  $L$ th impulse response value of  $A_0(z^L)$  is non-zero. Consequently, a linear-phase filter  $A(z^L)$  of order  $N_A$  requires roughly only  $N_A/(2L)$  multiplications and  $N_A/L$  additions. In this way, substantial overall savings can be obtained as compared to the conventional direct-form structures.

## 2.4. Design

Filters are typically designed in the minimax (Chebyshev) sense or least-squares sense, or possibly combinations thereof, see [24–26]. The goal of this chapter is to demonstrate that the complexity (number of multiplications and additions) can be reduced when using the two-rate based structures instead of regular structures. This will be done by designing both filter classes to meet the same specification and then comparing the resulting complexities<sup>4</sup>. To this end, the selection of approximation type is irrelevant, as long as one uses the same for both filter classes. In this chapter, we use minimax design, but other designs can of course be used as well after some minor appropriate modifications.

For minimax design, the maximum of the modulus of an error function  $E(j\omega)$  is minimized. The error function is typically given as

$$E(j\omega) = W(\omega)[H(e^{j\omega}) - D(j\omega)], \quad \omega \in \Omega. \quad (12)$$

<sup>4</sup> Another type of comparison is to study the approximation error differences between two solutions having the same filter implementation complexity. However, such a comparison is appropriate when using two different design methods applied to the same filter class (structure) which does not apply here.



where  $D(j\omega)$  is a desired function to be approximated in the frequency band  $\Omega$  by the filter frequency response  $H(e^{j\omega})$ , whereas  $W(\omega)$  is a positive weighting function. A conventional FIR filter, with the frequency response in the form of (2), is then designed by solving the following approximation problem.

**Approximation problem:** Given  $N_H$ , find the unknowns  $h(n)$  and  $\delta$  to minimize  $\delta$  subject to

$$|E(j\omega)| \leq \delta. \quad (13)$$

For a linear-phase filter, we also have the additional symmetry constraints  $h(n) = h(N - n)$  or  $h(n) = -h(N - n)$ .

For a conventional filter, the problem above is a convex optimization problem which has a unique global optimum. It can be found using linear programming, see [27], or the more efficient McClellan-Parks-Rabiner algorithm given in [28]. In practice, one usually has a specification on the desired approximation error  $\delta$ , say  $\delta_e$ . The filter will meet this specification if  $\delta$  after the optimization satisfies  $\delta \leq \delta_e$ .

For the two-rate based filters, the design becomes more intricate because it contains cascaded and parallel subfilters. This means that the unknowns are not  $h(n)$  but instead  $f(n)$  and  $g(n)$ , in general, and  $a(n)$ ,  $b_0(n)$ , and  $g(n)$  when  $F(z)$  is realized as an FRM filter. Hence, conventional design methods can no longer be used. Moreover, due to the cascaded subfilters, we are now facing a nonlinear (nonconvex) optimization problem, which means that an overall globally optimum solution cannot be guaranteed. Nevertheless, if carefully designed, even a locally optimum solution for a two-rate based structure can be substantially less complex than the corresponding globally optimum direct-form structure. To ensure a good local optimum, the overall two-rate based filters are designed in three steps as explained below. Although  $F(z)$  should here be an FRM HB filter in order to achieve any savings, we will first explain the essential design steps in terms of a regular HB filter for the sake of simplicity. After that, the necessary modifications required for an FRM design will be pointed out.

#### 2.4.1. Basic Three-Step Design Procedure

Given the desired function

$$D(j\omega) = e^{-j\omega(N_G+N_F)/4} D_0(j\omega) \quad (14)$$

and bandwidth  $\omega \in [0, \omega_c]$ ,  $\omega_c < \pi$ , as well as a targeted approximation error  $\delta_e$ , perform the following three-step procedure for each combination of filter orders  $N_G$  and  $N_F$  around estimated required orders  $\hat{N}_G$  and  $\hat{N}_F$ :

- (1) Design the regular FIR filter  $G(z)$ , which gives  $G_0(z)$  and  $G_1(z)$  after polyphase decomposition. It is done by minimizing the maximum of  $|E_G(j\omega)|$  in the band  $\omega \in [0, \omega_c/2]$  where <sup>5</sup>

$$E_G(j\omega) = G(e^{j\omega}) - e^{-j\omega N_G/2} D_0(j2\omega). \quad (15)$$

<sup>5</sup> For a right-band specification, the band for  $G(z)$  is  $\omega \in [\omega_c/2, \pi/2]$ .



- (2) Design a regular lowpass HB FIR filter  $F(z)$ , which gives  $F_0(z)$  and  $F_1(z) = z^{-(D_F-1)/2}$ ,  $D_F = N_F/2$ , after polyphase decomposition. It is done by minimizing the maximum of  $|E_F(j\omega)|$  in the band  $\omega \in [\pi - \omega_c/2, \pi]$ , where <sup>6</sup>,

$$E_F(j\omega) = F(e^{j\omega}). \quad (16)$$

- (3) Use  $F_0(z)$ ,  $G_0(z)$ , and  $G_1(z)$  obtained above as the initial solution in a further nonlinear optimization routine that solves the approximation problem stated in (13). If the resulting approximation error  $\delta$  is smaller than  $\delta_e$  after the optimization, store the result.

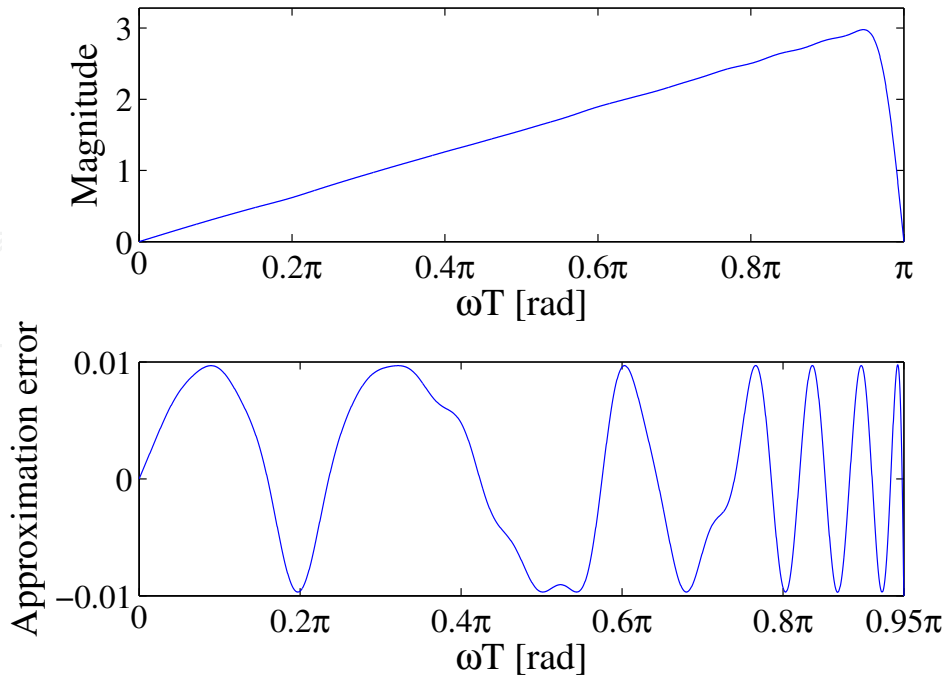
The estimated orders required,  $\hat{N}_G$  and  $\hat{N}_F$ , can be found by separately designing  $G(z)$  and  $F(z)$  to approximate their respective desired functions (as given in Steps 1 and 2, respectively) with the same tolerance as the overall targeted error, i.e.,  $\delta_e$  (or similar as in [17–19]). As the bandwidth of  $G(z)$  is always below  $\pi/2$ , its order is typically below 12 for approximation errors down to some  $-100$  dB, provided a smooth function is targeted, like a differentiator or integrator. Hence, the value of  $\hat{N}_G$  is readily found by designing  $G(z, d)$  for all low-order filters, using conventional techniques, and then set  $\hat{N}_G$  to the lowest one for which the approximation error  $|E_G(j\omega)|$  is below  $\delta_e$ . As to the lowpass HB filter  $F(z)$ , the value  $\hat{N}_F$  can be found via well-known formulas for order estimation, see [20, 21], and a few designs around the estimated value.

Regarding the designs, the problems in Steps 1 and 2 are convex, and thus have unique global optima, provided they are formulated in accordance with the approximation problem stated earlier in this section. These problems can be solved using any regular solver for such problems. As  $F(z)$  is a linear-phase filter, it can alternatively be designed using the efficient McClellan-Parks-Rabiner algorithm given in [28]. The problem in Step 3 is nonlinear because of the cascaded subfilters. In the examples of this chapter, we use the general-purpose nonlinear-optimization routine *fminimax* in MATLAB together with the real-rotation theorem, see [29], to solve the problem. The real-rotation theorem states that minimizing  $|f|$  is equivalent to minimizing  $\Re\{fe^{j\Theta}\}$ ,  $\forall \Theta \in [0, 2\pi]$ . The optimization problem is then solved with  $\omega$  and  $\Theta$  discretized to dense enough grids. A few hundred and 10–20 points, respectively, are typically sufficient in practice.

#### 2.4.2. Modifications When Using an FRM Filter $F(z)$

When  $F(z)$  is an FRM filter, we can use essentially the same design steps as outlined above. However, a difference is that  $F(z)$  is now realized in terms of the two subfilters  $A(z^P)$  and  $B_0(z)$  or, equivalently,  $F_0(z)$  is now realized in terms of the three subfilters  $A_0(z^L)$ ,  $B_{00}(z)$  and  $B_{01}(z)$ . This means that three parameters,  $\hat{N}_A$ ,  $\hat{N}_B$ , and  $\hat{L}$ , instead of only one parameter,  $\hat{N}_F$ , need to be estimated. Given the same approximation error and band edges as before,  $F(z)$  as well as  $\hat{N}_A$ ,  $\hat{N}_B$ , and  $\hat{L}$ , can be obtained as outlined in [7]. It is noted here that the design of  $F(z)$  in Step 2 now corresponds to a nonconvex problem due to cascaded subfilters in the FRM approach. In [7], this is solved via initial linear optimizations and further nonlinear optimization, similar to the approach given above for the two-rate based structure.

<sup>6</sup> For a right-band specification, a highpass filter  $F(z)$  is designed instead, in the stopband  $\omega \in [0, \omega_c/2]$ .



**Figure 2.** Magnitude response and approximation error of the two-rate and FRM based filter in Example 1. Throughout this chapter, we have used  $T = 1$  for simplicity.

## 2.5. Examples

Consider a first-degree differentiator with the desired function [4]

$$D(j\omega) = e^{-j\omega(N_G+N_F)/4}j\omega \quad (17)$$

in the frequency region  $\omega T \in [0, \omega_c]$ ,  $0 < \omega_c < \pi$ . This function can be approximated by a Type III linear-phase FIR filter, i.e., by a filter of even order and with an anti-symmetric impulse response, see [4].

*Example 1:*  $\omega_c = 0.95\pi$ , and  $\delta_e = 0.01$  (−40 dB). Using a conventional differentiator, the specification is met by a 60th-order filter which requires 30 multiplications and 59 additions in an implementation. Using instead the two-rate and FRM based approach, with  $L = 5$ , we can meet the specification with filter orders 22, 18, and 2, for  $A(z)$ ,  $B_0(z)$ , and  $G(z)$ , respectively. The corresponding overall realization requires 17 multiplications and 31 additions. Thus, multiplication and addition savings of 43% and 47%, respectively, are achieved. The savings are however dependent of the bandwidth  $\omega_c$  as will be illustrated below in Example 2. As always when using linear-phase FRM filters, the price to pay is a somewhat increased delay, and a few more delay elements. In this example, the delay is increased from 30 to 32 samples whereas the number of delay elements is increased from 60 to 64. The increase is thus only 7%. The overall filter frequency response is plotted in Fig. 2.

*Example 2:* Figure 3 shows the number of multiplications required for the conventional direct-form filter and the two-rate based filter, both approximating first-degree Type III differentiators with approximation errors of  $\delta = 0.01, 0.001, 0.0001$  (−40, −60, −80 dB).

As the plots reveal, the complexity savings using the two-rate based filter is increased substantially when the bandwidth approaches  $\pi$ . The break-even point is somewhere around  $\omega_c = 0.8\pi$  from which the savings increase approximately linearly with increasing bandwidth. In the region between  $0.8\pi$  and  $0.98\pi$ , the savings go from around zero up to some 65%. Similar savings are obtained also for the number of additions as it is proportional to the number of multiplications. Again, a price to pay for the arithmetic complexity reductions is a moderate increase of the delay and number of delay elements, typically between some 5% and 20%.

From the results in [17, 22], the number of multiplications required for a regular Type III differentiator can be estimated as

$$\hat{M}_{\text{regular}} = \pi \frac{0.810[-\log_{10}(\delta_c)]^{0.919}}{\pi - \omega_c}. \quad (18)$$

For the two-rate based differentiators, we have instead from [17]

$$\hat{M}_{\text{tworate}} = \pi \frac{0.884[-\log_{10}(\delta_c)]^{0.852}}{\pi - 0.956\omega_c}. \quad (19)$$

Comparing the two expressions, we see that the main difference is the multiplicative constant 0.956 in front of  $\omega_c$  in the latter expression. This explains why the savings increase with increasing bandwidth, as illustrated in Fig. 3.

## 2.6. Generalization to $M > 2$

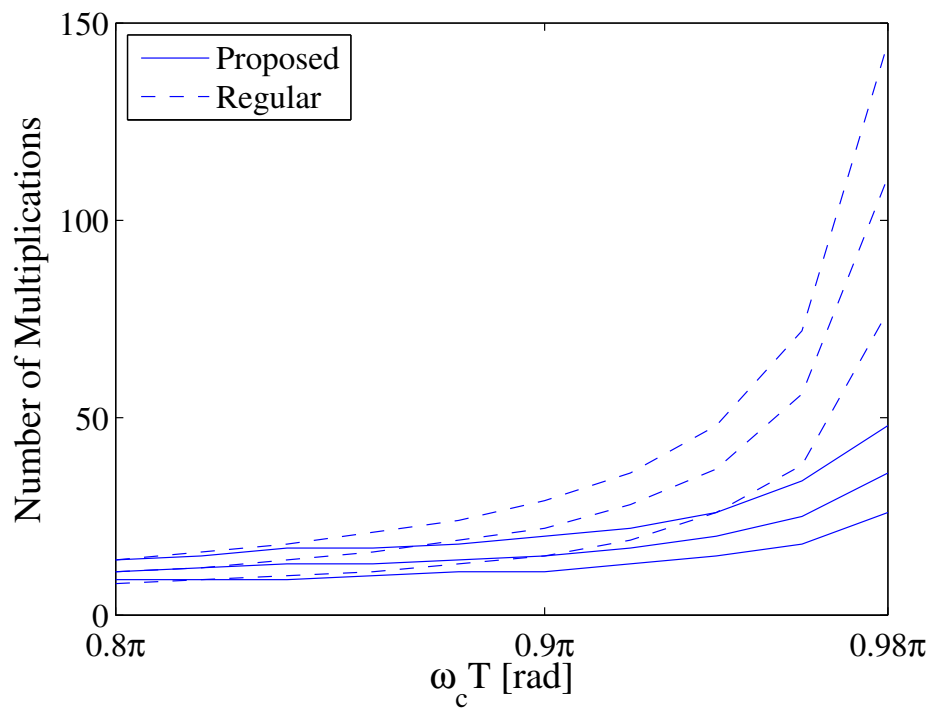
The two-rate based scheme can readily be extended to the one depicted in Fig. 4(a) where the interpolation factor is an arbitrary integer  $M$ . Here, the basic principle is thus to first interpolate with  $M$  via the interpolation filter  $F(z)$ . Then the actual function is again approximated by  $G(z)$ . Finally, downsampling by  $M$  occurs. Using multi-rate theory, one finds again that this structure has the LTI system equivalent seen in Fig. 4(b). That is, the overall transfer function is

$$H(z) = F_0(z)G_0(z) + \sum_{m=1}^{M-1} z^{-1} F_m(z)G_{M-m}(z), \quad (20)$$

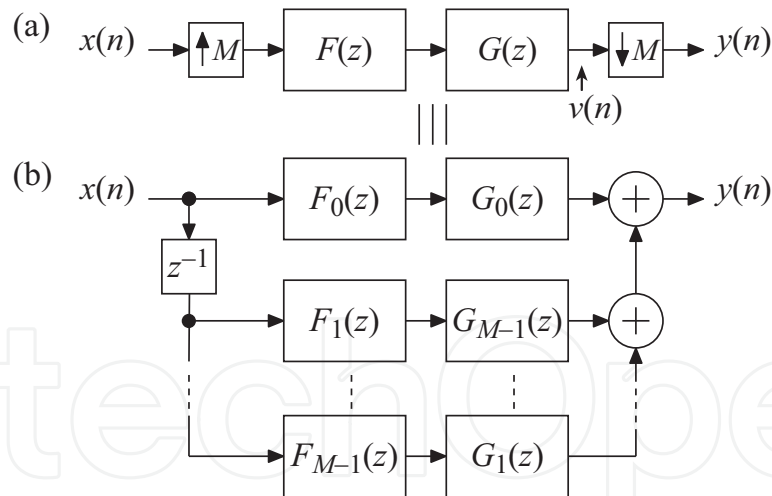
where  $F_m(z)$  and  $G_m(z)$  are polyphase components of  $F(z)$  and  $G(z)$  in the polyphase representations

$$F(z) = \sum_{m=0}^{M-1} z^{-m} F_m(z^M), \quad G(z) = \sum_{m=0}^{M-1} z^{-m} G_m(z^M). \quad (21)$$

Using an  $M$ th-band interpolation filter  $F(z)$ , also the generalized scheme is appropriate for left-band and right-band systems. It has turned out though that the case with  $M = 2$  typically is the most efficient choice which is why that case has been considered in detail in this section. This is because the additional cost of  $F(z)$  exceeds the additional savings of  $G(z)$  when going from  $M = 2$  to  $M > 2$ . This in turn is due to the fact that the complexity of  $G(z)$  is already very low for  $M = 2$ . A more detailed discussion on this is found in [17].



**Figure 3.** Number of multiplications versus bandwidth  $\omega_c T$  for the conventional direct-form filter (dashed line) and proposed two-rate based filter (solid line), for a first-degree Type III differentiator with the approximation errors  $\delta_c = 0.01, 0.001, 0.0001$ . Throughout this chapter, we have used  $T = 1$  for simplicity.



**Figure 4.** (a) Two-rate approach with arbitrary  $M$ . (b) Equivalent single-rate realization.

### 3. Two-rate based structure for mid-band systems

This section extends the results to mid-band systems which target the region  $\omega \in [\omega_{c1}, \omega_{c2}]$ ,  $0 < \omega_{c1} < \omega_{c2} < \pi$ . Example applications include fractional-degree differentiators and integrators, see [30–33]. For later discussions, we define the don't-care bands  $\Delta_1$  and  $\Delta_2$  as

$$\Delta_1 = \omega_{c1}, \quad \Delta_2 = \pi - \omega_{c2}. \quad (22)$$

In principle, we can again make use of the scheme in Fig. 4 with a lowpass filter  $F(z)$  but it is not efficient for mid-band systems. This is because the filter  $G(z)$  then needs to approximate the desired function in the band between  $\omega_{c1}/M = \Delta_1/M$  and  $\omega_{c2}/M = (\pi - \Delta_2)/M$ . Although this implies that the width of the upper don't-care band of  $G(z)$  is increased substantially to roughly  $(M - 1)\pi/M$  instead of the original  $\Delta_2 = \pi - \omega_{c2}$ , its lower don't-care band,  $\Delta_1/M = \omega_{c1}/M$ , becomes  $M$  times narrower. This means that the complexity of  $G(z)$  may thereby even increase, not decrease. In the left-band case, this is not a problem as there is no don't-care band to the left.

The width of both the lower and the upper don't-care bands of  $G(z)$  can be increased by using a bandpass filter  $F(z)$  instead of a lowpass filter. This also means that we have to use  $M > 2$ . Again, it appears that the most efficient case is for the lowest possible  $M$  which is here  $M = 3$ . The reason for this is two-fold. First, odd values of  $M$  makes it possible to center the passband of  $G(z)$  around  $\pi/2$ , which maximizes the minimum of its lower and upper don't-care bands. Second, the complexity of  $F(z)$  alone reduces with reduced  $M$ , in accordance with the discussion in [17] for the left-band case. In addition, the use of  $M = 3$  instead of  $M > 3$ , makes it possible to double the amount of sparsity of  $F(z)$ , and thus its efficiency, by expressing it as a periodic filter.

Here,  $G(z)$  is to approximate  $D(j\omega M - j(K - 1)\pi)$  in the frequency region  $\omega \in [\omega_{c1}^{(G)}, \omega_{c2}^{(G)}]$  where

$$\omega_{c1}^{(G)} = (K - 1)\pi/M + \Delta_1/M, \quad \omega_{c2}^{(G)} = K\pi/M - \Delta_2/M, \quad (23)$$

with  $K$  being an appropriately chosen odd integer. For  $M = 3$ , one should use  $K = -1$ . After the downsampling by  $M$ , the above region is mapped to the targeted region  $\omega \in [\omega_{c1}, \omega_{c2}]$ . Further,  $F(z)$  is to approximate  $M$  in the same region as that of  $G(z)$  and zero in the corresponding image bands created in the upsampling. Hence,  $F(z)$  is here a bandpass filter with passband and stopband edges at

$$\omega_{c1}^{(F)} = \omega_{c1}^{(G)}, \quad \omega_{c2}^{(F)} = \omega_{c2}^{(G)} \quad (24)$$

and

$$\omega_{s1}^{(F)} = (K - 1)\pi/M - \Delta_1/M, \quad \omega_{s2}^{(F)} = K\pi/M + \Delta_2/M, \quad (25)$$

respectively. Moreover, with  $M = 3$  and  $K = -1$ ,  $F(z)$  is a symmetric bandpass filter centered on  $\pi/2$ . Consequently, it can be expressed as

$$F(z) = 3P(z^2) \quad (26)$$

where  $P(z)$  is a unity-gain-passband third-band highpass filter. The polyphase decomposition of  $P(z)$  is then  $P(z) = 1/3 + z^{-1}P_1(z^3) + z^{-2}P_2(z^3)$  which leads to  $F(z) = 1 + 3z^{-2}P_1(z^6) + 3z^{-4}P_2(z^6)$  and the polyphase components

$$F_0(z) = 1, \quad F_1(z) = 3z^{-1}P_2(z^2), \quad F_2(z) = 3P_1(z^2). \quad (27)$$

A filter  $F(z)$  of the form above requires roughly only one third of the complexity of a general filter of the same order.

	$N_H$	$N_G$	$N_F$	$D_H$	DE	Mult	Add
Regular	124	-	-	62	124	125	124
Two-rate, regular $F(z) = P(z^2)$	126	10	372	63	126	73	134
Two-rate, FRM $F(z) = P(z^2)$	140	10	412	70	140	41	59

**Table 2.** Results of Example 3.

### 3.1. Complexity savings

As opposed to the case of linear-phase overall filters considered in Section 2, we can here achieve complexity savings without using additional FRM techniques. The reason is two-fold. First, as seen above,  $F(z) = P(z^2)$  is already sparse. Second, as a mid-band system is often a nonlinear-phase system, the filter coefficient are not symmetric. By using the two-rate based structure, symmetry can partially be utilized as  $F(z)$  is a symmetric filter whereas only the low-order  $G(z)$  is unsymmetric. As to the sparsity, the degree of sparseness can be increased by realizing  $P(z)$  as an FRM third-band filter. Details are given in [18].

### 3.2. Examples

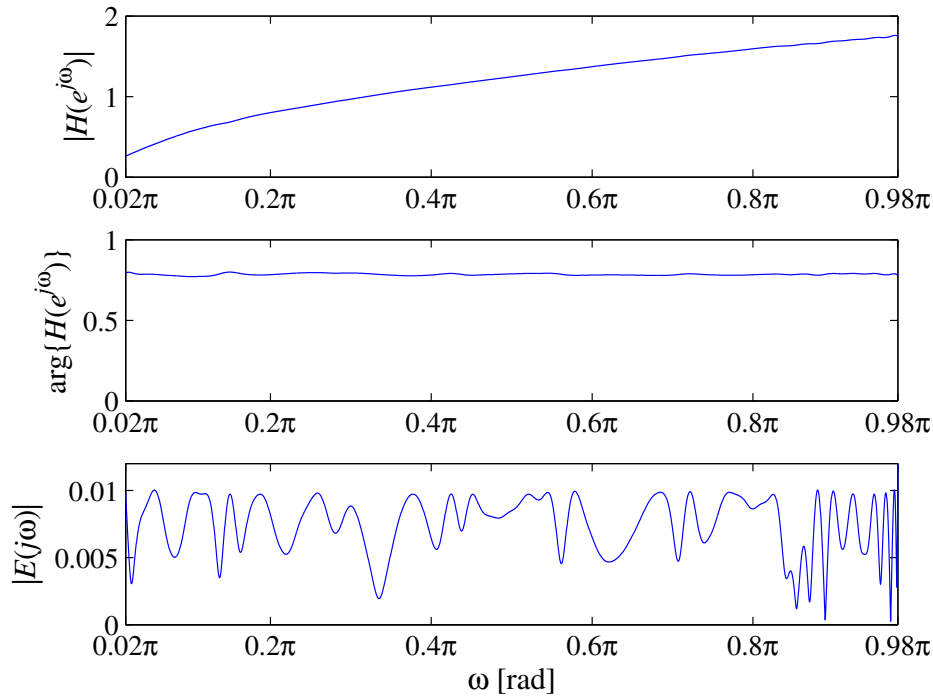
*Example 3:* Consider the approximation of a fractional-degree differentiator with the desired function  $D(j\omega) = e^{-j\omega(N_G+N_F)/4}(j\omega)^{0.5}$  in the frequency band  $\omega \in [0.02\pi, 0.98\pi]$  and for an approximation error of  $\delta_e = 0.01$ . Figure 5 shows the frequency response and approximation error of the two-rate based design. The filter has been designed using essentially the same three-step procedure described earlier, but after minor appropriate modifications, as detailed in [18]. Table 2 gives the results for the conventional direct-form realization and for the two-rate based realizations, both with a sparse regular bandpass filter and a sparse FRM bandpass filter. The quantity  $D_H$  denotes the integer part of the group delay whereas DE denotes the number of delay elements. As seen from the table, substantial savings are achieved using the two-rate based structures, especially when the FRM technique is also utilized. As usual when using the FRM technique, one has to pay a price in a somewhat increased delay. It is also noted that the savings increase/decrease with increased/decreased bandwidth (decreased/increased width of the don't-care bands). This is in line with the basic two-rate based scheme and it was exemplified earlier in Example 2.

## 4. Multi-function systems

In this section, we will discuss the extension to the realization of multifunction systems. The two-rate based approach is even more efficient for such systems as the same  $F(z)$ , and thus the same  $F_0(z)$ , is shared between all functions. We will illustrate this for Farrow-structure based (see [34]) variable fractional-delay (VFD) filters. As an example will reveal, the two-rate based structure offers dramatic complexity reductions in this application, even without using the additional FRM approach. However, incorporating the FRM approach, further complexity savings are obtained.

### 4.1. Variable fractional-delay filters

Variable fractional-delay filters find applications in many different contexts like interpolation, resampling, delay estimation, and signal reconstruction, see [35–40].



**Figure 5.** Fractional-degree differentiator responses in Example 3 using the two-rate based structure with an FRM bandpass filter targeting  $\omega_{c1} = 0.02\pi$  and  $\omega_{c2} = 0.98\pi$ .

The VFD filter, with a transfer function  $H(z, d)$ , is for  $z = e^{j\omega}$  to approximate the ideal VFD filter frequency response

$$D(j\omega, d) = e^{-j\omega(D_H + d)} \quad (28)$$

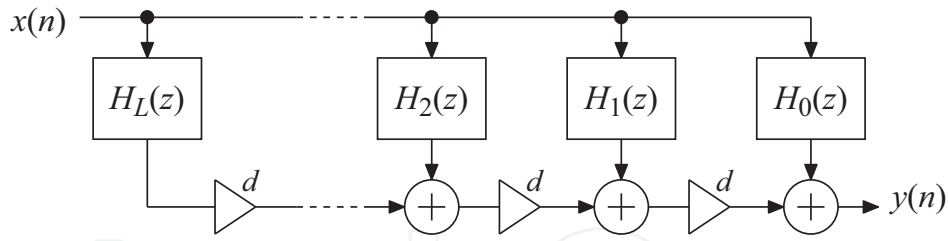
where  $D_H$  is a fixed delay which usually is an integer or an integer plus a half. Further,  $d$  is the fractional delay. The ideal response should be approximated in the band  $\omega \in [0, \omega_c]$ ,  $0 < \omega_c < \pi$ , and for all fractional delays  $d \in [-0.5, 0.5]$  meaning that a whole sampling period (interval) is covered. In general, the sampling period is  $T$ , but we have used  $T = 1$  in this chapter for simplicity.

Using the Farrow structure,  $H(z, d)$  is expressed in the form

$$H(z, d) = \sum_{k=0}^L d^k H_k(z). \quad (29)$$

where  $H_k(z)$  are fixed subfilters which, essentially, realize the weighted differentiators  $e^{-j\omega D_H} \times (-j\omega)^k / k!$ . This follows immediately by truncating the Taylor series expansion of  $e^{-j\omega d}$ , see [41]. Further, when there are no restrictions on the fixed part of the delay, it is possible and efficient to use linear-phase subfilters  $H_k(z)$ , thus with symmetric or antisymmetric impulse responses. We then have  $D_H = N_H/2$ , and the following two different cases. When  $G_k(z)$  are of even order  $N_G$ , they are of Type I (Type III) for even (odd) values of  $k$ . This results in an integer  $D_H$ . In the odd-order case,  $G_k(z)$  are instead of Type II (Type IV) for even (odd) values of  $k$ . In this case,  $D_H$  is an integer plus a half. In both





**Figure 6.** Farrow structure realizing the VFD filter transfer function in (29).

cases, the impulse responses are symmetric (anti-symmetric) for even (odd) values of  $k$ , thus  $g_k(n) = (-1)^k g_k(N_G - n)$ .

Figure 6 shows the regular Farrow structure realizing (29). As seen, the problem amounts to realizing the  $L + 1$  differentiator functions with ideal responses  $(-j\omega T)^k/k!$ . In other words, it essentially corresponds to the realization of a multi-function system, although, in this case, the partial outputs are finally combined via the FD multiplications to form only one output. Using now the two-rate based approach introduced in Section 2.2, each  $H_k(z)$  is realized as

$$H_k(z) = 2F_0(z)G_{k0}(z) + z^{-(D_F+1)/2}G_{k1}(z) \quad (30)$$

where  $F_0(z)$  and  $z^{-(D_F-1)/2}$  are again the polyphase components of a linear-phase HB interpolation filter  $F(z)$  with a passband gain of two and delay  $D_F$ , whereas  $G_{k0}(z)$  and  $G_{k1}(z)$  are the polyphase components of the subfilters  $G_k(z)$ . This follows from sandwiching the filter  $F(z)G(z, d)$  between the upsampler and downsampler, where  $G(z, d)$  approximates an FD filter in the region  $[0, \omega_c/2]$ . That is,

$$G(z, d) = \sum_{k=0}^L d^k G_k(z). \quad (31)$$

The overall realization is shown in Fig. 7. It is noted that  $F(z)$  again can be realized using the FRM approach in order to further reduce the complexity, as demonstrated in [19]. In this case,  $F_0(z)$  is again realized as in Fig. 1(c) or (d).

## 4.2. Design examples

*Example 4:* We consider the design of a VFD filter with a bandwidth of  $\omega_c = 0.9\pi$ . The filter has been designed using essentially the same three-step procedure described earlier. More details are given in [19]. Tables 3 and 4 summarize the results where the number of multiplications and additions covers all fixed subfilters assuming appropriate use of direct-form and transposed direct-form realizations. In addition,  $L$  general multipliers and adders are needed for implementing the FD multiply-and-add chain, but this is required in all VFD filter structures. Further, the NRMS and  $\delta_{gd}$  values given in the tables indicate the normalized root-mean square error and maximum group-delay error as defined by (33) and (36), respectively, in [42], whereas  $\delta_e$  denotes the maximum of the modulus of the complex error. Further, DE denotes the number of delay elements. It is seen from the table that the two-rate based structure is considerably more efficient than the regular Farrow structure.

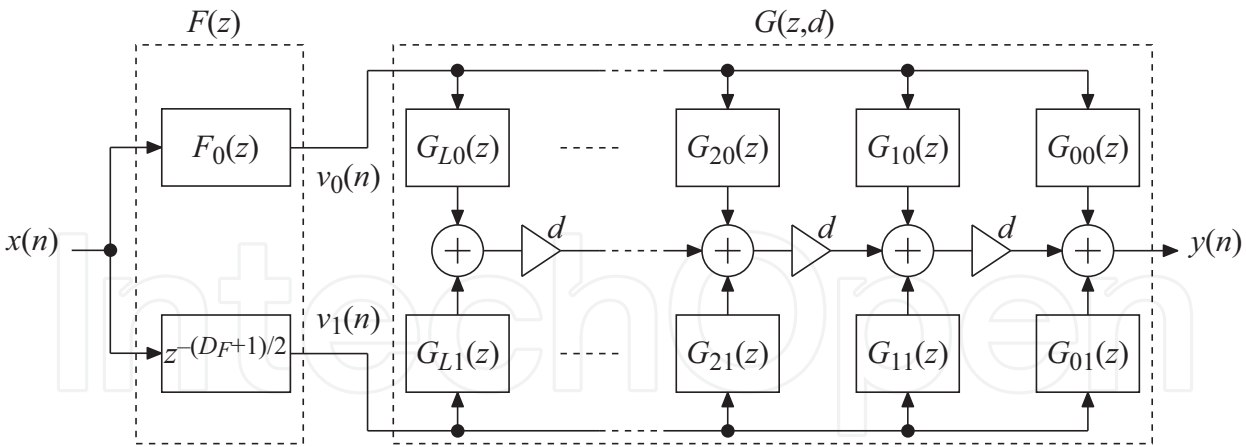


Figure 7. Two-rate based structure realizing the VFD filter transfer function in (29) with  $H_k(z)$  as in (30).

Linear-Phase	$L$	$N_H$	$N_E$	$N_F$	$N_A$	$N_B$	$P$
Reg. Farrow, WLS [44]	7	73	n/a	n/a	n/a	n/a	n/a
Hybrid, WLS[42]	7	117	n/a	n/a	n/a	n/a	n/a
Simplified [43]	9	73	n/a	n/a	n/a	n/a	n/a
Two-rate based	7	75	12	138	n/a	n/a	n/a
Two-rate based with FRM	7	87	12	162	46	26	3

Table 3. Results of Example 4.

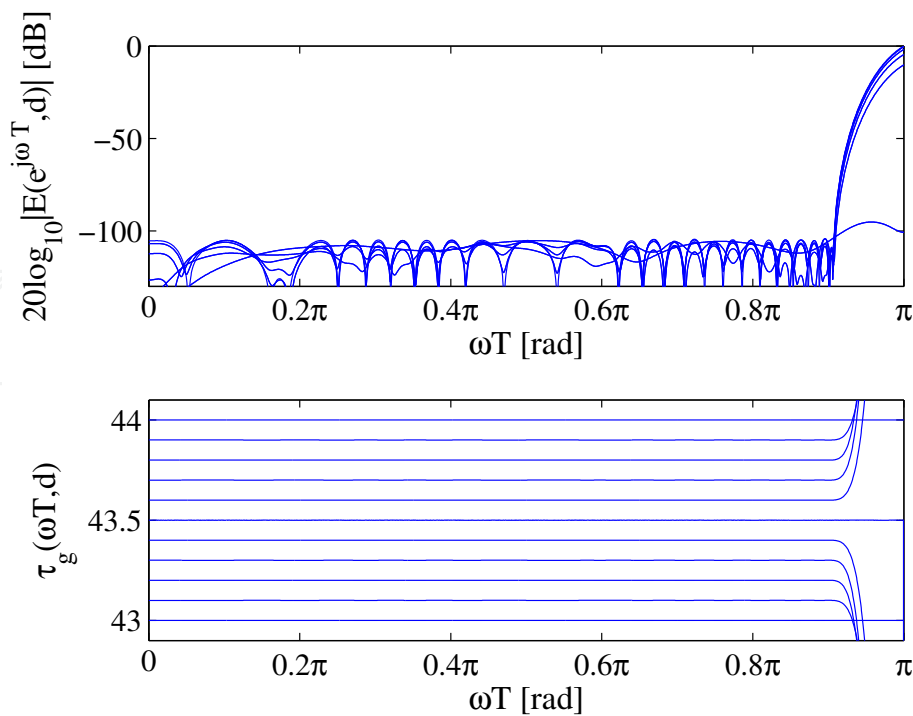
Linear-Phase	$D_H$	DE	Mult	Add	NRMS (%)	$\delta_e$ [dB]	$\delta_{gd}$
Reg. Farrow, WLS [44]	36.5	73	191	374	0.00023	−100.04	0.000446
Hybrid, WLS[42]	58.5	117	148	285	0.00021	−100.21	0.000395
Simplified [43]	36.5	73	91	158	not reported	−102.14*	not reported
Two-rate based	37.5	75	80	149	0.00019	−102.96	0.000318
Two-rate based with FRM	43.5	87	71	129	0.00019	−104.15	0.000263

Table 4. Results of Example 4. \*Magnitude and phase delay errors.

It is also more efficient than two alternative approaches whose results are also included in the table, namely for the hybrid structure in [42] and the structure in [43] which meets roughly the same specification. It is also seen that the extended structure that utilizes the FRM technique offers further complexity reductions. The price to pay in this case is however a slight increase of the delay and delay elements, but the figures are still considerably smaller than for the structure in [42]. Compared with the regular Farrow structure in [44] and the one in [43], one has to pay the moderate price of a delay and delay element increase of some 3% using the basic structure in Fig. 7(b) and 19% using the extended structure incorporating the FRM approach.

5. Multiple-constant multiplication techniques for the subfilter implementations

This section will discuss implementation details, design trade-offs, and comparisons when the multiplications in the filters are implemented using multiple constant multiplication (MCM) techniques, which realize a number of constant multiplications using only shifts,

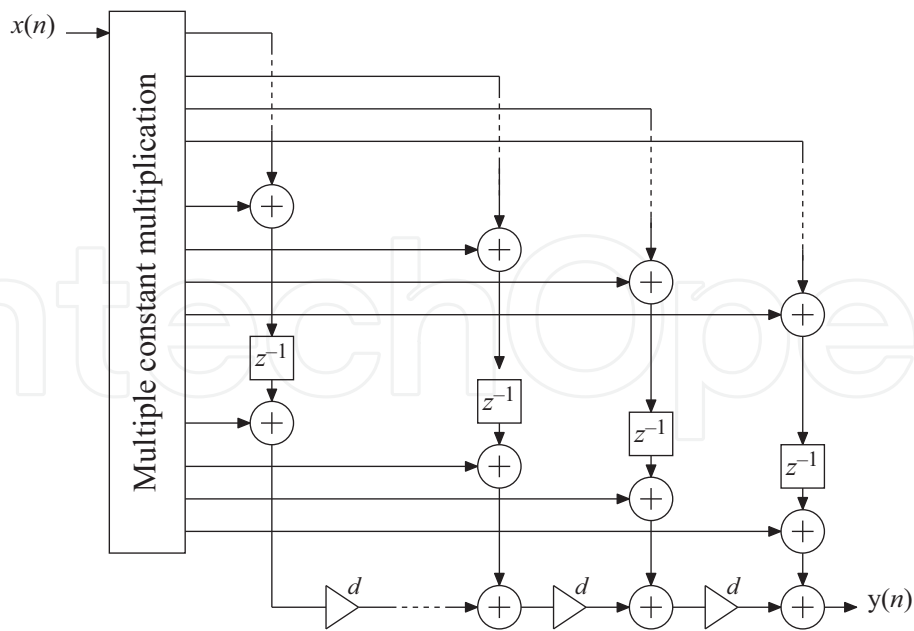


**Figure 8.** Two-rate and FRM based VFD filter in Example 4. Error and group delay responses for 11 evenly distributed values of  $d$  between  $-0.5$  and  $0.5$ .

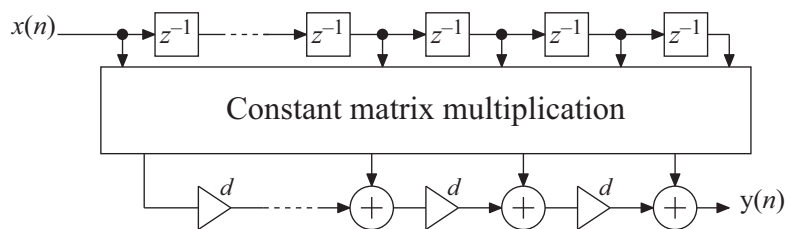
adders and subtracters. The situation is different here though, than for the most commonly considered transposed direct-form filter realization, as the proposed structures consist of cascaded subfilters. This section will therefore elaborate on these issues and provide design examples. The focus here is on VFD filters using the two-rate based structure without the additional FRM approach.

For dedicated hardware implementations, one can take advantage of MCM techniques to reduce the implementation cost. Multiplications by constant coefficients can be performed using adders, subtracters, and shifts. As adders and subtracters have approximately the same implementation complexity we will refer to both as adders. Efficient realization of constant multiplications is an active research area and much effort has been focused on the case where one input data is multiplied by several constant coefficients. This problem has mainly been motivated by single-rate FIR filters, where for a transposed direct form FIR filter the input is multiplied by several coefficients, see [45–49]. The resulting implementation of several multiplications is denoted multiplier block, as in [45].

Work has also been done for sampling rate change with an integer factor in [50] and rational factor in [51], where it was shown that FIR filters in parallel can be implemented either using one multiplier block or by using a constant matrix multiplication block, as in [52–55], with the first approach requiring more delay elements than the latter. As a Farrow filter also is composed of several FIR filters in parallel we have the same implementation alternatives here, not only the single multiplier block case as reported in [56]. This has been extensively discussed in [57]. In Fig. 9, the approach to implement the subfilters proposed in [56] is shown. This approach typically requires few additions for the multiplier block. However, a separate set of registers is required for each subfilter and the number of structural adders



**Figure 9.** Realization of Farrow filter using transposed direct form subfilters resulting in a multiplier block and several sets of registers.

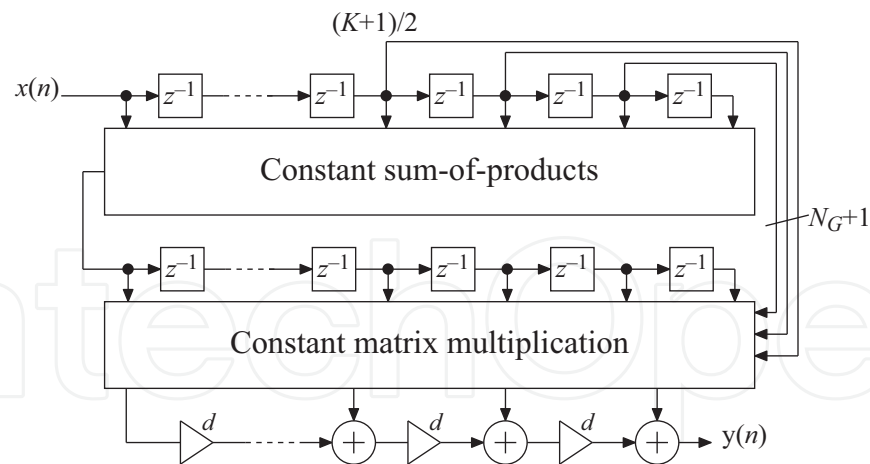


**Figure 10.** Realization of Farrow filter using direct form subfilters resulting in a constant matrix multiplication and a single set of registers.

is high. Alternatively, in Fig. 10, an approach based on the observation in [50] and further discussed in [57] is shown. Here, only one set of registers is required and the structural adders of the subfilters are merged into the matrix-vector multiplication.

The Farrow filter part of the two-rate based structure in Fig. 7 can be implemented similarly to what is shown in Figs. 9 or 10. For transposed direct form subfilters, as in Fig. 9, the corresponding structure would have two inputs, and, hence, result in a constant matrix multiplication. Using direct form subfilters, as in Fig. 10, requires two sets of registers, one for each input. For the HB filter it is convenient to use a direct form subfilter as the delayed input values are easily obtained from the registers. We note that the input to the lower branch subfilters in Fig. 7 is just a delayed version of the input, which is available from the upper branch subfilter  $F_0(z)$ . Therefore, it is possible to use the registers of the HB filter as registers for direct form subfilters. The resulting structure is illustrated in Fig. 11.

Naturally, it is also possible to use a transposed direct form HB filter and/or transposed direct form subfilters in the Farrow filter part. From a complexity point of view, a transposed direct form HB filter will have the same number of adders and registers. However, it will not be possible to share registers as shown in Fig. 11.



**Figure 11.** Realization of the filter structure in Fig. 7 using direct form subfilters resulting in a sum-of-products block and a constant matrix multiplication.

## 5.1. Example and comparisons

*Example 5:* We consider a specification where the bandwidth is  $0.9\pi$  and the modulus of the complex error should be below 0.0042. To meet this specification, the Farrow structure in Fig. 6, with subfilters jointly optimized as outlined in detail in [41], requires 45 fixed multipliers, 88 fixed adders, and 5 variable multipliers. The two-rate based structure in Fig. 7, with subfilters jointly optimized as detailed in [14], requires 30 fixed multipliers, 53 fixed adders, and 5 variable multipliers. Thus, in terms of number of multiplications and additions, the two-rate based structure is superior.

To refine the comparison when MCM techniques are applied we must quantize the filter coefficients. For a relative comparison, one can use simple rounding. We found that the original Farrow structure requires 11 bits to fulfil the requirements whereas the structure in Fig. 7 requires 13 fractional bits. The slightly larger number of bits for the two-rate approach is explained by the fact that a cascaded filter must meet the requirements which leads to a somewhat more stringent requirement on the subfilters, at least when simple rounding is used.

For the regular Farrow structure in Fig. 6, together with the realization in Fig. 9 a total of 33 adders are required for the multiplier block using the RAG-n algorithm in [45]. This is an optimal result since there are 33 different (odd) coefficients as discussed in [58], and, hence, there is no need to apply the slightly more efficient algorithms in [47–49]. Furthermore, 80 structural adders and 118 registers are required for the FIR subfilters. Further, five general multipliers and four additional adders are required (for both the regular Farrow and two-rate based filters). Alternatively, using the proposed structure in Fig. 10 the constant matrix multiplication can be realized using 96 adders using the algorithm in [53]. In addition, 26 structural adders are required. One observation is that separating the symmetric and anti-symmetric subfilters may reduce the complexity as some algorithms work better for fewer columns. If this is utilized, the number of adders can be reduced to 107 by applying the algorithm in [52] to the resulting two matrices and adding and subtracting the results. The number of registers is now decreased to 30, whereas the number of general multipliers and additional adders is constant.

For the two-rate based structure in Fig. 7, the HB filter requires 43 structural adders and 29 registers. The sum of products are realized by computing the corresponding multiplier

Filter structure	Mult Add Registers		
Farrow with transposed direct-form subfilters	5	117	118
Farrow with direct-form subfilters	5	111	30
Proposed two-rate based structure	5	85	35

**Table 5.** Results of Example 5.

block with RAG-n and transposing it. For the constant matrix multiplication, 38 adders are required. This number is not reduced by separating the symmetric and anti-symmetric subfilters. A total of six additional registers are required as well as the five general multipliers and four additional adders.

The results are summarized in Table 5. It is seen that the two-rate based structure still has the lowest complexity for most implementation technologies as five registers will typically be less complex to implement than 26 adders. Furthermore, whereas the use of transposed direct form subfilters for the Farrow filter, as proposed in [50, 57], reduces the number of adders related to the multiplication, it is for the two-rate approach still more efficient to use direct form subfilters.

6. Conclusion

This chapter has reviewed recent two-rate based structures and their design for obtaining efficient wide-band FIR systems. Left-band, right-band, and mid-band systems, as well as single-function and multi-function systems, were covered. Several design examples were given, for differentiators and VFD filters (a special case of multi-function systems), revealing dramatic complexity savings for wide-band specifications. More details can be found in [12–19].

Author details

Håkan Johansson and Oscar Gustafsson

Division of Electronics Systems, Department of Electrical Engineering, Linköping University, Sweden

References

[1] M. Renfors and Y. Neuvo. The maximum sampling rate of digital filters under hardware speed constraints. *IEEE Trans. Circuits Syst.*, 28(3):196–202, Mar. 1981.

[2] A. Fettweis. On assessing robustness of recursive digital filters. *Eur. Trans. Telecomm. Relat. Technol.*, 1(2):103–109, Mar.–Apr. 1990.

[3] Y. C. Lim. Frequency-response masking approach for the synthesis of sharp linear phase digital filters. *IEEE Trans. Circuits, Syst.*, CAS-33(4):357–364, Apr. 1986.

[4] T. Saramäki. Finite impulse response filter design. In S.K. Mitra and J.F. Kaiser, editors, *Handbook for Digital Signal Processing*, chapter 4, pages 155–277. New York: Wiley, 1993.



- [5] T. Saramäki, Y. C. Lim, and R. Yang. The synthesis of half-band filter using frequency-response masking technique. *IEEE Trans. Circuits Syst. II*, 42(1):58–60, Jan. 1995.
- [6] P. S. R. Diniz, L. C. R. de Barcellos, and S. L. Netto. Design of high-resolution cosine-modulated transmultiplexers with sharp transition band. *IEEE Trans. Signal Processing*, 52(5):1278–1288, May 2004.
- [7] H. Johansson. Two classes of frequency-response masking linear-phase FIR filters for interpolation and decimation. *Circuits, Syst., Signal Processing*, 25(2):175–200, Apr. 2006.
- [8] R. Bregovic, Y. C. Lim, and T. Saramäki. Frequency-response masking-based design of nearly perfect-reconstruction two-channel FIR filterbanks with rational sampling factors. *IEEE Trans. Circuits Syst I: Regular Papers*, 55(7):2002–2012, July 2008.
- [9] Y. Wei and Y. Lian. Frequency-response masking filters based on serial masking schemes. *Circuits, Syst., Signal Processing*, 29(1):7–24, Feb. 2010.
- [10] J. Yli-Kaakinen and T. Saramäki. An efficient algorithm for the optimization of FIR filters synthesized using the multistage frequency-response masking approach. *Circuits, Syst., Signal Processing*, 30(1):157–183, Feb. 2011.
- [11] N. P. Murphy, A. Krukowski, and I. Kale. Implementation of a wide-band integer and fractional delay element. *Electron. Lett.*, 30(20):1658–1659, 1994.
- [12] G. Jovanovic-Dolecek and J. Diaz-Carmona. One structure for wide-bandwidth and high-resolution fractional delay filter. In *Proc. IEEE Int. Conf. Electr. Circuits Syst.*, 2002.
- [13] E. Hermanowicz. On designing a wideband fractional delay filter using the Farrow approach. In *Proc. XII European Signal Processing Conf.*, Vienna, Austria, Sept. 6–10 2004.
- [14] E. Hermanowicz and H. Johansson. On designing minimax adjustable wideband fractional delay FIR filters using two-rate approach. In *Proc. European Conf. Circuit Theory Design*, Cork, Ireland, Aug. 29–Sept. 1 2005.
- [15] H. Johansson, O. Gustafsson, K. Johansson, and L. Wanhammar. Adjustable fractional-delay FIR filters using the Farrow structure and multirate techniques. In *Proc. IEEE Asia Pacific Conf. Circuits Syst.*, 2006.
- [16] J. Diaz-Carmona and G. Jovanovic-Dolecek. Frequency-based optimization design for fractional delay FIR filters with software-defined radio applications. *Int. J. Digital Multimedia Broadcasting*, 53(6):1–6, June 2010.
- [17] Z. U. Sheikh and H. Johansson. A class of wide-band linear-phase FIR differentiators using a two-rate approach and the frequency-response masking technique. *IEEE Trans. Circuits Syst. I: Regular papers*, 58(8):1827–1839, Aug. 2011.



- [18] Z. U. Sheikh and H. Johansson. Efficient wide-band FIR LTI systems derived via multi-rate techniques and sparse bandpass filters. *IEEE Trans. Signal Processing*, to appear, 60(7):3859–3863, July 2012.
- [19] H. Johansson and E. Hermanowicz. Two-rate based low-complexity variable fractional-delay FIR filter structures. *IEEE Trans. Circuits Syst. I: Regular papers*, to appear, 2012.
- [20] J. F. Kaiser. Nonrecursive digital filter design using  $I_0$ -sinh window function. In *Proc. Int. Symp. Circuits Syst.*, volume 3, pages 20–23, Apr. 1974.
- [21] K. Ichige, M. Iwaki, and R. Ishii. Accurate estimation of minimum filter length for optimum FIR digital filters. *IEEE Trans. Circuits Syst. II*, 47(10):1008–1016, Oct. 2000.
- [22] Z. U. Sheikh, A. Eghbali, and H. Johansson. Linear-phase FIR digital differentiator order estimation. In *Proc. European Conf. Circuit Theory Design*, Linköping, Sweden, Aug. 29–31 2011.
- [23] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice Hall, 1993.
- [24] L. Rabiner, J. McClellan, and T. Parks. FIR digital filter design technique using weighted-Chebyshev approximation. *IEEE Proc.*, 63(4):595–610, Apr. 1975.
- [25] G. Mollova. Compact formulas for least-squares design of digital differentiators. *Electron. Lett.*, 35(20):1695–1697, Sept. 1999.
- [26] J. J. Shyu, S. C. Pei, and Y. D. Huang. Least-squares design of variable maximally linear FIR differentiators. *IEEE Trans. Signal Processing*, 57(11):4568–4573, Nov. 2009.
- [27] S. G. Nash and A. Sofer. *Linear and Nonlinear Programming*. McGraw-Hill, 1996.
- [28] J. H. McClellan, T. W. Parks, and L.R. Rabiner. A computer program for designing optimum FIR linear phase digital filters. *IEEE Trans. Audio Electroacoust.*, AU-21:506–526, Dec. 1973.
- [29] T. W. Parks and C. S. Burrus. *Digital Filter Design*. John Wiley and Sons, 1987.
- [30] Y. Q. Chen and K. L. Moore. Discretization schemes for fractional-order differentiators and integrators. *IEEE Trans. Circuits Syst. I.*, 49:363–367, Mar. 2002.
- [31] R. S. Barbosa, J. A. T. Machado, and M. F. Silva. Time-domain design of fractional differintegrators using least-squares. *Signal Processing*, 86(10):2567–2581, Oct. 2006.
- [32] C. C. Tseng and S.-L. Lee. Design of fractional order digital differentiator using radial basis function. *IEEE Trans. Circuits Syst. I: Regular Papers*, 57(7):1708–1718, July 2010.
- [33] B. T. Krishna. Studies on fractional order differentiators and integrators: A survey. *Signal Processing*, 91(3):386–426, Mar. 2011.
- [34] C. W. Farrow. A continuously variable delay element. In *Proc. IEEE Int. Symp., Circuits, Syst.*, volume 3, pages 2641–2645, Espoo, Finland, June 7–9 1988.

- [35] F. M. Gardner. Interpolation in digital modems—Part I: Fundamentals. *IEEE Trans. Comm.*, 41(3):502–508, Mar. 1993.
- [36] T. I Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine. Splitting the unit delay—Tools for fractional delay filter design. *Signal Processing Mag.*, 13(1):30–60, Jan. 1996.
- [37] S. R. Dooley and A. K. Nandi. On explicit time delay estimation using the Farrow structure. *Signal Processing*, 72:53–57, Jan. 1999.
- [38] H. Johansson and P. Löwenborg. Reconstruction of nonuniformly sampled bandlimited signals by means of digital fractional delay filters. *IEEE Trans. Signal Processing*, 50(11):2757–2767, Nov. 2002.
- [39] M. Olsson, H. Johansson, and P. Löwenborg. Time-delay estimation using Farrow-based fractional-delay FIR filters: filter approximation vs. estimation errors. In *Proc. XIV European Signal Processing Conf.*, Florence, Italy, Sept. 4–8 2006.
- [40] S. Tertinek and C. Vogel. Reconstruction of nonuniformly sampled bandlimited signals using a differentiator-multiplier cascade. *IEEE Trans. Circuits Syst. I: Regular papers*, 55(8):2273–2286, Sept. 2008.
- [41] H. Johansson and P. Löwenborg. On the design of adjustable fractional delay FIR filters. *IEEE Trans. Circuits Syst. II*, 50(4):164–169, Apr. 2003.
- [42] T.-B. Deng. Hybrid structures for low-complexity variable fractional delay filters. *IEEE Trans. Circuits Syst. I: Regular Papers*, 57(4):897–910, Apr. 2010.
- [43] J. Yli-Kaakinen and T. Saramäki. A simplified structure for FIR filters with an adjustable fractional delay. In *Proc. IEEE Int. Symp. Circuits Syst.*, New Orleans, USA, May 27–30 2007.
- [44] T.-B. Deng. Symmetric structures for odd-order maximally flat and weighted-least-squares variable fractional-delay filters. *IEEE Trans. Circuits Syst. I: Regular Papers*, 54(12):2718–2732, Dec. 2007.
- [45] A. G. Dempster and M. D. Macleod. Use of minimum-adder multiplier blocks in FIR digital filters. *IEEE Trans. Circuits Syst. II*, 42(9):569–577, Sept. 1995.
- [46] R. I. Hartley. Subexpression sharing in filters using canonic signed digit multipliers. *IEEE Trans. Circuits Syst. II*, 43(10):677–688, Oct. 1996.
- [47] Y. Voronenko and M. Püschel. Multiplierless multiple constant multiplication. *ACM Trans. Algorithms*, 2006.
- [48] O. Gustafsson. A difference based adder graph heuristic for multiple constant multiplication problems. In *Proc. IEEE Int. Symp. Circuits Syst.*, pages 1097–1100, New Orleans, USA, May 27–30, 2007.
- [49] L. Aksoy, E.O. Günes, and P. Flores. Search algorithms for the multiple constant multiplications problem: Exact and approximate. 34(5):151–162, August 2010.

- [50] O. Gustafsson and A. G. Dempster. On the use of multiple constant multiplication in polyphase fir filters and filter banks. In *Proc. Nordic Signal Processing Symp.*, pages 53–56, Espoo, Finland, June 9–11, 2004.
- [51] O. Gustafsson and H. Johansson. Efficient implementation of FIR filter based rational sampling rate converters using constant matrix multiplication. In *Proc. Fortieth Asilomar Conf. Signals, Systems and Computers ACSSC '06*, pages 888–891, 2006.
- [52] A. G. Dempster, O. Gustafsson, and J. O. Coleman. Towards an algorithm for matrix multiplier blocks. In *Proc. European Conf. Circuit Theory Design*, Krakow, Poland, Sept. 1–4, 2003.
- [53] M. D. Macleod and A. G. Dempster. A common subexpression elimination algorithm for low-cost multiplierless implementation of matrix multipliers. *Electronics Lett.*, 40(11):651–652, May 2004.
- [54] N. Boullis and A. Tisserand. Some optimizations of hardware multiplication by constant matrices. *IEEE Trans. Computers*, 54(10):1271–1282, Oct. 2005.
- [55] L. Aksoy, E. Costa, P. Flores, and J. Monteiro. Optimization algorithms for the multiplierless realization of linear transforms. *ACM Trans. Design Automation Electronic Syst.*, vol. 17, no. 1, article 3, Jan. 2012.
- [56] A. G. Dempster and N. P. Murphy. Efficient interpolators and filter banks using multiplier blocks. *IEEE Trans. Signal Processing*, 48(1):257–261, Jan. 2000.
- [57] M. Abbas, O. Gustafsson, and H. Johansson. On the fixed-point implementation of fractional-delay filters based on the farrow structure. *IEEE Trans. Circuits Syst I: Regular Papers*, 2012. to appear.
- [58] O. Gustafsson. Lower bounds for constant multiplication problems. *IEEE Trans. Circuits Syst. II*, 54(11):974–978, Nov. 2007.