CIRCUITS SYSTEMS SIGNAL PROCESSING
VOL. 26, NO. 2, 2007, PP. 115–146

DOI: 10.1007/s00034-005-0704-8

POLYNOMIAL-BASED INTERPOLATION FILTERS—PART I: FILTER SYNTHESIS*

Jussi Vesma[1] *and Tapio Saramäki*[2]

Abstract. This paper introduces a generalized design method for polynomial-based interpolation filters. These filters can be implemented by using a modified Farrow structure, where the fixed finite impulse response (FIR) sub-filters possess either symmetrical or anti-symmetrical impulse responses. In the proposed approach, the piecewise polynomial impulse response of the interpolation filter is optimized directly in the frequency domain using either the minimax or least mean square criterion subject to the given time domain constraints. The length of the impulse response and the degree of the approximating polynomial in polynomial intervals can be arbitrarily selected. The optimization in the frequency domain makes the proposed design scheme more suitable for various digital signal processing applications and enables one to synthesize interpolation filters for arbitrary desired and weighting functions. Most importantly, the interpolation filters can be optimized in a manner similar to that of conventional linear-phase FIR filters.

Key words: Polynomial-based interpolation, Farrow structure, interpolation, sampling rate conversion.

1. Introduction

In digital signal processing applications *general interpolation filters* (or simply interpolation filters) are utilized to evaluate new sample values at arbitrary points between the existing discrete-time samples. The terms “fractional delay (FD) filter” and “interpolator” are also used in this context (see, e.g., [7] and [11]). These kinds of interpolation filters that are considered in this paper have one

* Received July 4, 2005; revised May 7, 2006; published online April 26, 2007. This work was supported by the Academy of Finland project no. 44876 (Finnish Centre of Excellence Program (2000–2005)) and the Nokia Foundation. Some parts of the material in this paper are presented in Ref. [20].

[1] Nokia Corporation, Turku, FIN-20521 Finland. E-mail: jussi.vesma@nokia.com

[2] Institute of Signal Processing, Tampere University of Technology, P.O. Box 553, FIN-33101 Tampere, Finland. E-mail: ts@cs.tut.fi

continuous-valued input parameter, called the *fractional interval*, which can be used to control the time instant for the output sample.

The discrete-time impulse response of the interpolation filter is determined by the value of the fractional interval and, in typical applications, it is usually time varying. This makes it difficult to analyze and design interpolation filters with all possible values of the fractional interval. A commonly used solution to overcome this problem is to model the interpolation filter with an analog/digital system where a continuous-time signal is reconstructed by using a digital-to-analog (D/A) converter and a reconstruction filter. This reconstructed signal is then sampled at the desired time instants to obtain the interpolated sample values [3], [12]. A widely used structure for the general interpolation filters is based on the use of finite impulse response (FIR) digital filters having a fractional delay (see, e.g., [11]). In order to be able to control the time instants for the output samples, a large number of *fractional delay* (FD) FIR filters with various delay values have to be synthesized, and the filter coefficients have to be stored in a lookup table.

Because of the efficient implementation structure, one of the most interesting class of interpolation filters is the *polynomial-based* interpolation filters. This class consists of filters having a piecewise polynomial impulse response. The most attractive feature of these filters is that they can be efficiently implemented using the Farrow structure [6]. This discrete-time filter structure consists of parallel FIR filters with fixed coefficient values. The desired time instant for the interpolated output sample can be easily controlled by properly weighting the output samples of these FIR filters by the corresponding fractional interval.

The design methods for polynomial-based interpolation filters can be roughly divided into two different classes. The first class consists of the time domain methods, where the approximating polynomial is fitted to the discrete-time samples. The best known time domain methods are based on the conventional Lagrange and B-spline interpolations, where the filter coefficients for the Farrow structure are easily available in the closed form [5], [14]. The problem is that the only design parameter is the degree of the approximating polynomial. Furthermore, if there are frequency components rather close to half the sampling rate, then the approximation provided by these interpolation methods becomes poor.

In the second class of design methods for polynomial-based interpolation filters, the coefficients of the Farrow structure are optimized directly in the frequency domain [6], [8], [11], [17], [19]. This approach is much more flexible, and it enables one to design interpolation filters with better filtering characteristics than those obtained by the conventional time domain interpolation methods. For example, in [11] Laakso et al. introduced a combined frequency/time domain synthesis method and, in [6], Farrow suggested a least-mean-square optimization of the polynomial-based FD filters. These methods do not allow separate optimization of the passband and stopband regions of the interpolation filter.

This paper introduces a synthesis scheme for polynomial-based interpolation filters where the frequency domain characteristics of the input signal can be better taken into account than in the above-mentioned methods. The proposed technique is based on the fact that if the impulse response of the reconstruction filter in the above-mentioned analog/digital model is a piecewise polynomial, then the polynomial coefficients uniquely determine the coefficients of the corresponding Farrow structure. Due to this fact, the design of the overall interpolator can be converted to that of the reconstruction filter.

One desirable feature of the proposed technique is that the frequency response of the reconstruction filter can be expressed in terms of the polynomial coefficients in a manner similar to the expression of a linear-phase FIR filter in terms of its coefficient values. This makes the optimization of the overall system very straightforward.

Depending on the application, the reconstruction filter may possess many passbands and stopbands, the desired amplitude and weight functions can be selected arbitrarily for each band, and the length of the filter and the degree of the interpolation can be chosen independently. The actual optimization can be performed either in the minimax sense using linear programming or in the least-mean-square sense using techniques similar to those proposed for synthesizing linear-phase FIR filters. The resulting interpolation filters provide significantly better frequency domain performances than those based on the use of the existing design methods.

The outline of this paper is as follows. The general interpolation filter is defined in Section 2 along with the analog/digital system that is used to model these filters. The Farrow structure for polynomial-based interpolation filters is reviewed in Section 3. Sections 4 and 5 present the proposed design method for polynomial-based interpolation filters. It is shown how the impulse responses of the fixed FIR filters in the original Farrow structure are formed by utilizing certain kinds of basis functions enabling more efficient implementation than the original Farrow structure. The resulting structure is the so-called modified Farrow structure introduced by Vesma and Saramäki in [20]. This structure consists of a given number of fixed odd-order linear-phase FIR filters that alternatively possess symmetrical and anti-symmetrical impulse responses, thereby properly exploiting the coefficient symmetries enables one to reduce the number of multiplies by a factor of 2 when compared to the original Farrow structure. Furthermore, it is shown how the filter coefficients are optimized by using either the minimax or least-mean-square criterion. Some examples are included in Section 6 to compare the benefits provided by the proposed method over conventional methods. The concluding remarks are presented in Section 7.

2. Interpolation filters

This section presents a general discrete-time interpolation filter and shows how this filter can be studied by using the hybrid analog/digital model.

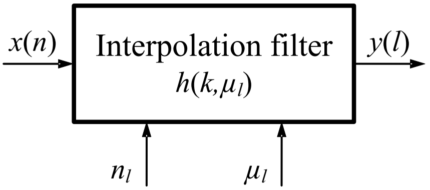


Figure 1. General interpolation filter with impulse response $h(k, \mu_l)$, input signal $x(n)$, and interpolated output samples $y(l)$. The input parameters n_l and μ_l are used to determine the time instant $t = t_l$ for the output samples $y(l)$ according to equation (1).

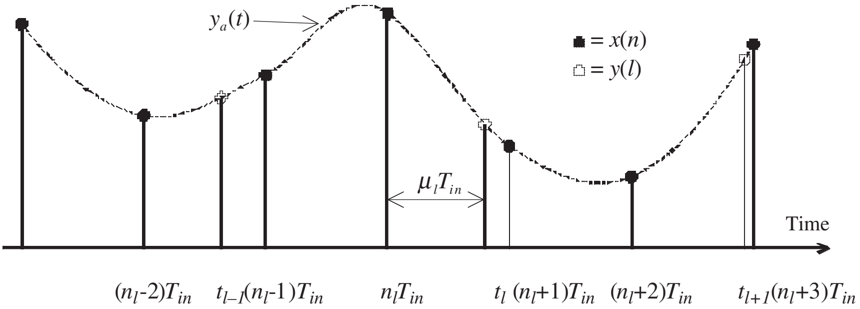


Figure 2. Interpolation in the time domain. The approximating continuous-time signal $y_a(t)$ (dashed line) is generated by the interpolation process based on the existing discrete-time samples $x(n)$. The interpolated samples are then given by $y(l) = y_a(t_l)$.

2.1. General interpolation filter

Figure 1 shows a simplified block diagram for the general interpolation filter. The interpolation process in the time domain is illustrated in Figure 2. The idea is first to form an approximating continuous-time signal $y_a(t)$ based on the existing discrete-time samples $x(n)$ with a sampling interval of $T_{in} = 1/F_{in}$ and, then, to sample $y_a(t)$ at the desired time instants denoted by t_l to obtain the interpolated sample values $y(l) = y_a(t_l)$.

The parameters n_l and μ_l are used to determine the time instant t_l for the lth output sample as follows:

$$t_l = (n_l + \mu_l)T_{in}, \tag{1}$$

where n_l is an integer and μ_l, which is called the *fractional interval*, is specified in the interval $0 \leq \mu_l < 1$. Given the time instant for the output sample t_l, these parameters are determined as [7]

$$n_l = \lfloor t_l / T_{in} \rfloor \tag{2a}$$

and

$$\mu_l = t_l/T_{in} - \lfloor t_l/T_{in} \rfloor , \tag{2b}$$

where $\lfloor x \rfloor$ stands for the integer part of x.

This contribution concentrates mostly on generating the output sample of the interpolation filter by using the following convolution:

$$y(l) = \sum_{k=-N/2}^{N/2-1} x(n_l - k)h(k, \mu_l), \tag{3}$$

where N, the filter length, is assumed to be an even integer and $h(k, \mu_l)$ is the time-varying impulse response of the interpolation filter. Note that these impulse response values depend on μ_l.

2.2. Interpolation filters based on the fractional delay filters

A widely used approach to design and implement interpolation filters is to use FIR [or infinite impulse response (IIR)] filters with a delay that is a fraction of the sample interval T_{in}. When using this design approach, the fractional interval μ_l is first quantized using K uniformly spaced quantization levels (see [18] for the effects of the quantization). Then, K different *fractional delay* (FD) filters having fractional delay[3] values of $\gamma = k/K$ for $k = 0, 1, \ldots, K - 1$ are synthesized (see, e.g., [11]). In order to have an interpolation filter with an adjustable fractional interval μ_l, the coefficients of the FD filters are stored in a lookup table.

The disadvantages of this FD filter approach are that the size of the lookup table is usually large and the overall system is difficult to analyze because the impulse responses of the FD filters are determined separately for each value of μ_l. In order to avoid the problems of the FD filter approach, the next subsection reviews how the interpolation generally can be interpreted as a hybrid analog/digital model.

2.3. Hybrid analog/digital model for interpolation filters

Interpolation is basically a reconstruction problem, where the approximating continuous-time signal $y_a(t)$ is reconstructed based on the existing discrete-time samples $x(n)$. Therefore, a useful way to model interpolation filters is to use *the hybrid analog/digital model* depicted in Figure 3 [3], [12].

In this system, the continuous-time signal $y_a(t)$ is reconstructed by using a D/A converter and a reconstruction filter $h_a(t)$. The interpolated output sample $y(l)$ is then obtained by sampling $y_a(t)$ at $t_l = (n_l + \mu_l)T_{in}$. If it is assumed that the noncausal reconstruction filter $h_a(t)$ is zero outside the interval $-NT_{in}/2 \leq t <$

[3] In order to have a positive delay in FD filters, the fractional delay γ has a different sign than the fractional interval μ_l and they are related to each other through $\gamma = 1 - \mu_l$.

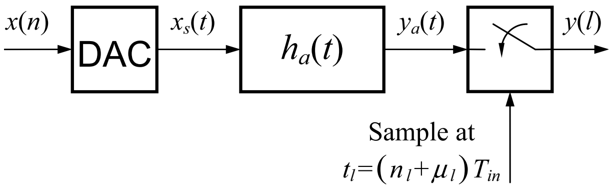


Figure 3. Hybrid analog/digital model for the interpolation filter.

$NT_{in}/2$, then the lth interpolated output sample can be expressed as (see, e.g., [16])

$$y(l) = y_a(t_l) = \sum_{k=-N/2}^{N/2-1} x(n_l - k)h_a((k + \mu_l)T_{in}). \tag{4}$$

By comparing equation (3) and (4), it is seen that the continuous-time impulse response of the reconstruction filter $h_a(t)$ and the time-varying impulse response of the interpolation filter $h(k, \mu_l)$ are related to each other as follows:

$$h(k, \mu_l) = h_a((k + \mu_l)T_{in}) \text{ for } k = -N/2, -N/2 + 1, \ldots, N/2 - 1. \tag{5}$$

The hybrid analog/digital model of Figure 3 converts the design of the overall interpolation filter with time-varying impulse response $h(k, \mu_l)$ to that of the reconstruction filter with time-invariant response $h_a(t)$. Therefore, the analog model can be utilized in the design process of interpolation filters as follows:

(1) Find the continuous-time impulse response of the reconstruction filter $h_a(t)$ so that its frequency response, denoted by $H_a(j2\pi f)$, approximates the given desired response according to some criterion.[4]

(2) Develop an efficient digital implementation structure based on the convolution of equation (3) and the relation between $h(k, \mu_l)$ and $h_a(t)$, as given by equation (5). What kind of implementation structures can be developed depends strongly on how $h_a(t)$ is generated.

3. Polynomial-based interpolation filters

The impulse response of a polynomial-based interpolation filter is expressible by means of piecewise polynomials. The advantage of filters of this kind is that they can be efficiently implemented using the filter structure introduced by Farrow [6]. This section shows how the Farrow structure can be derived by starting with the above-mentioned hybrid analog/digital model.

[4] In the sequel, $h_a(t)$ and $H_a(j2\pi f)$ are called the impulse response and the frequency response of the interpolation filter, respectively.

3.1. Definition of polynomial-based interpolation filter

For a polynomial-based interpolation filter, the impulse response $h_a(t)$ is expressed in each interval of length T_{in} by means of a polynomial as follows:

$$h_a((k+\mu_l)T_{in}) = \sum_{m=0}^{M} \hat{c}_m(k)\mu_l^m \tag{6}$$

for $k = -N/2, -N/2+1, \ldots, N/2-1$ and for $\mu_l \in [0, 1)$. Here, the $\hat{c}_m(k)$'s are the polynomial coefficients for the kth interval of length T_{in} and M is the degree of the polynomials. It is assumed that $h_a(t)$ is nonzero only for $-NT_{in}/2 \leq t < NT_{in}/2$.

3.2. The Farrow structure

By using the hybrid analog/digital model of Figure 3, the digital implementation structure for polynomial-based interpolation filters can be derived by substituting the impulse response $h_a(t)$, as given by equation (6), into equation (4), yielding

$$y(l) = y_a(t_l) = \sum_{m=0}^{M} v_m(n_l)\mu_l^m, \tag{7}$$

where

$$v_m(n_l) = \sum_{k=-N/2}^{N/2-1} x(n_l-k)\hat{c}_m(k) \tag{8}$$

are the output samples of the $M+1$ parallel FIR filters of length N with coefficient values $\hat{c}_m(k)$. In the causal case, the transfer functions of these FIR filters are given by

$$\hat{C}_m(z) = \sum_{k=0}^{N-1} \hat{c}_m(k-N/2)z^{-k}. \tag{9}$$

The corresponding filter structure, the Farrow structure [6], is shown in Figure 4. The main advantage of the Farrow structure is that all the filter coefficients directly related to the polynomial coefficients of the impulse response $h_a(t)$ are fixed. The only changeable parameters are the fractional interval μ_l and n_l, which depend on the lth output sampling instant. Also, the control of μ_l is easier during the computation than in the implementation based on the use of FD FIR filters. Furthermore, the resolution of μ_l is limited only by the precision of the arithmetic, not by the size of the memory or lookup table.

Note that the approximating signal $y_a(t)$ is also a piecewise polynomial in each input sample interval, as suggested by equation (7), and the polynomial coefficients $v_m(n_l)$ are obtained from the output of the FIR filters. Overall, the

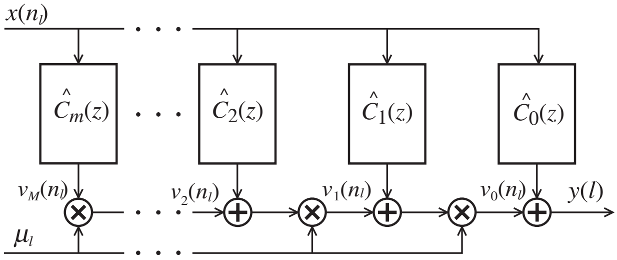


Figure 4. The Farrow structure for a polynomial-based interpolation filter.

multiplication by μ_l in the Farrow structure corresponds to the sampling of $y_a(t)$ in the analog model of Figure 3 at $t = t_l = (n_l + \mu_l)T_{in}$.

Note also that in the sampling rate conversion applications, the Farrow structure offers good anti-imaging properties, making it a suitable structure for cases where the sampling rate is increased. However, it does not offer good anti-aliasing properties for applications where the sampling rate is decreased (decimation). A better solution for decimation is the transposed Farrow structure, where the piecewise polynomial impulse response is determined not for the input but for the output sampling intervals [2], [9], [21]. Therefore, the transposed Farrow structure offers the same attenuation in the aliasing bands as the original structure in the imaging bands if the same polynomial coefficients are used.

4. Proposed generalized filter class

The main disadvantages of the classical Lagrange and B-spline interpolation methods are that the polynomial coefficients are derived in the time domain without utilizing the frequency domain information of the input signal and that there are very few adjustable parameters, resulting in a small number of design alternatives. On the other hand, synthesis techniques based on the use of the FD filters together with a frequency domain optimization suffer from the drawback that the optimization has to be performed separately for each value of the fractional interval μ_l. To overcome these problems, this paper introduces a general-purpose frequency domain optimization method for polynomial-based interpolation filters. This method utilizes the hybrid analog/digital model of Section 2 and enables one to select arbitrarily the length of the impulse response, the degree of the polynomials, and the desired and weighting functions in the frequency domain.

This section starts by constructing the generalized interpolation filter with a piecewise polynomial impulse response. After that, it is shown how the frequency response of this filter can be expressed in a simple form, enabling a straightforward optimization in the frequency domain using either the minimax or the least-mean-square criterion. Finally, a modified Farrow structure is introduced.

4.1. Impulse response for the generalized interpolation filter

The impulse response of the proposed noncausal interpolation filter, denoted by $h_a(t)$, is required to meet the following conditions:

(1) $h_a(t)$ is nonzero for $-NT_{in}/2 \le t < NT_{in}/2$.
(2) N is an even integer (for the case where N is odd, see [1]).
(3) $h_a(t)$ is a piecewise polynomial of degree M in each interval $nT_{in} \le t < (n+1)T_{in}$ for $n = -N/2, -N/2+1, \ldots, N/2-1$.
(4) $h_a(t)$ is symmetrical, that is, $h_a(-t) = h_a(t)$ except for the time instants $t = kT_{in}$ for $k = -N/2, -N/2+1, \ldots, -1$ and $k = 1, 2, \ldots, N/2$.

The desired $h_a(t)$ meeting conditions 1, 2, and 3 can be conveniently generated with the aid of the following polynomials:

$$f(m,t) = \begin{cases} \left(\dfrac{2t}{T_{in}} - 1\right)^m & \text{for } 0 \le t < T_{in} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

as

$$h_a(t) = \sum_{n=-N/2}^{N/2-1} \sum_{m=0}^{M} c_m(n) f(m, t - nT_{in}), \tag{11}$$

where the $c_m(n)$'s are the unknown polynomial coefficients. The first four polynomials $f(m,t)$ for $m = 0, 1, 2,$ and 3 are shown in Figure 5. By using the substitution $t = (n + \mu_l)T_{in}$ for $n = -N/2, -N/2+1, \ldots, N/2-1$, $h_a(t)$ can be also expressed by

$$h_a((n+\mu_l)T_{in}) = \sum_{m=0}^{M} c_m(n)(2\mu_l - 1)^m. \tag{12}$$

By properly determining the corresponding unknowns $c_m(n)$, any polynomial of degree M can be generated in the interval $nT_{in} \le t < (n+1)T_{in}$.

What is left is to determine $h_a(t)$ to also meet condition 4. This is achieved by requiring that

$$c_m(n) = (-1)^m c_m(-n-1) \tag{13}$$

for $m = 0, 1, \ldots, M$ and $n = 0, 1, \ldots, N/2-1$. This condition halves the number of unknowns and enables one to rewrite $h_a(t)$ with the aid of the following

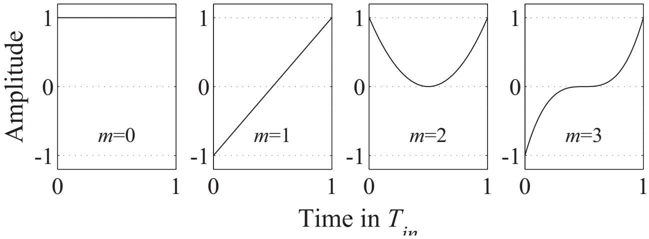


Figure 5. Polynomials $f(m, t)$ for $m = 0, 1, 2$, and 3.

basis functions:

$$g(n, m, t) = (-1)^m f(m, t + (n+1)T_{in}) + f(m, t - nT_{in})$$

$$= \begin{cases} (-1)^m \left(\dfrac{2(t + (n+1)T_{in})}{T_{in}} - 1 \right)^m & \text{for } -(n+1)T_{in} \\ & \le t < -nT_{in} \\ \left(\dfrac{2(t - nT_{in})}{T_{in}} - 1 \right)^m & \text{for } nT_{in} \le t < (n+1)T_{in} \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

as

$$h_a(t) = \sum_{n=0}^{N/2-1} \sum_{m=0}^{M} c_m(n) g(n, m, t). \tag{15}$$

As illustrated in Figure 6 for the $n = 1$ and $m = 3$ case, each resulting basis function $g(n, m, t)$ is characterized by the following properties:

(1) $g(n, m, t) = g(n, m, -t)$ for $-(n+1)T_{in} < t < -nT_{in}$.
(2) $g(n, m, t)$ is defined at $t = -(n+1)T_{in}$ and $t = nT_{in}$, but not at $t = -nT_{in}$ and $t = (n+1)T_{in}$.

Hence, all the basis functions are symmetrical around $t = 0$ except for the points $t = \pm nT_{in}$ and $t = \pm(n+1)T_{in}$. The exception is the $n = 0$ case, where the value of $g(0, m, t)$ at $t = 0$ is $(-1)^m$. Hence, the overall impulse response $h_a(t)$ is symmetrical around $t = 0$ except for the time instants $t = kT_{in}$ for $k = -N/2$, $-N/2 + 1, \ldots, -1$ and $k = 1, 2, \ldots, N/2$. Various time domain conditions at these points that are useful in practical applications will be considered in detail in Subsection 5.1.

Figure 7 shows how to construct the overall impulse response $h_a(t)$ for $N = 8$ and $M = 3$ according to equation (15). The weighted basis functions $c_m(n)g(n, m, t)$ for $m = 0, 1, 2$, and 3 are shown in Figures 7a–d, respectively. In each of these figures there are four weighted basis functions for which $n = 0, 1, 2$, and 3. The resulting overall impulse response $h_a(t)$ as shown in Figure 7e is then

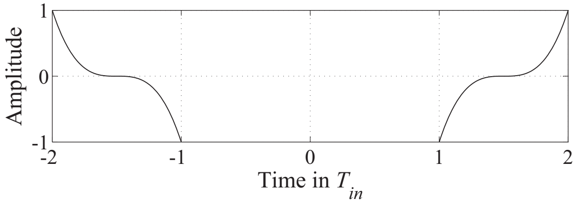


Figure 6. The basis function $g(n, m, t)$ for $n = 1$ and $m = 3$.

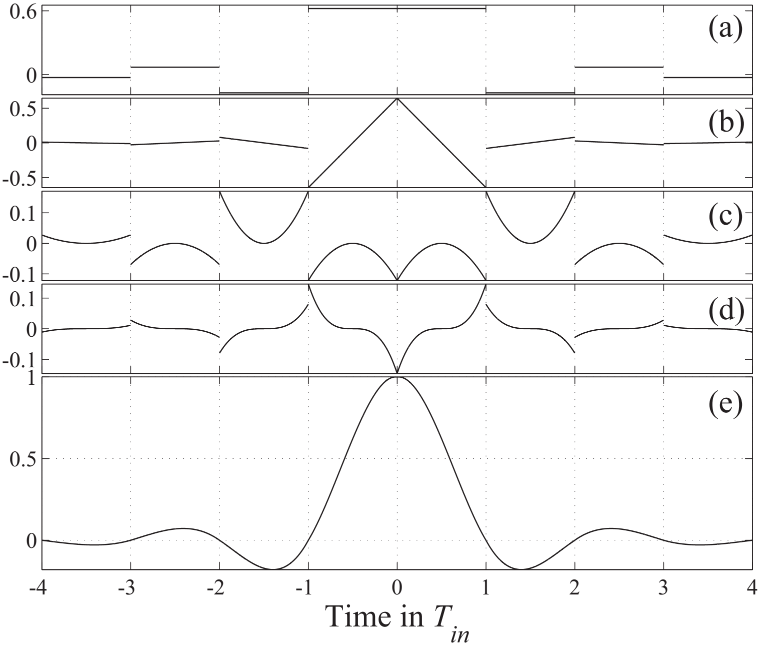


Figure 7. Construction of the overall impulse response $h_a(t)$ for $N = 8$ and $M = 3$. The weighted basis functions $c_m(n)g(n, m, t)$ for $n = 0$, 1, 2, and 3 and for (a) $m = 0$, (b) $m = 1$, (c) $m = 2$, and (d) $m = 3$. (e) The resulting impulse response $h_a(t)$.

generated by adding together all the weighted basis functions in Figures 7a–d. We point out that N, the length of the filter, and M, the degree of the polynomials, can be chosen independently, and $h_a(t)$ is not forced to take on the value of unity at $t = 0$ and the value of zero at $t = \pm T_{in}, \pm 2T_{in}, \ldots, \pm NT_{in}/2$ as for the Lagrange and B-spline interpolation filters.

4.2. Frequency response for the generalized interpolation filter

According to equation (15), the Fourier transform of $h_a(t)$ can be expressed as

$$H_a(f) \equiv H_a(j2\pi f) = \sum_{n=0}^{N/2-1} \sum_{m=0}^{M} c_m(n) G(n, m, f), \tag{16}$$

where $G(n, m, f)$ is the Fourier transform of the basis function $g(n, m, t)$. Because $g(n, m, t)$ is symmetrical around $t = 0$ except for the time instants $t = kT_{in}$ for $k = -N/2, -N/2+1, \ldots, -1$ and $k = 1, 2, \ldots, N/2$, $G(n, m, f)$ is real and is given by (see Appendix A)[5]

$$G(n, m, f) = \begin{cases} 2\cos(2\pi f(n+\frac{1}{2})T_{in}) \\ \quad \times \left[(-1)^{m/2} m! \Phi(m, f) + \dfrac{\sin(\pi f T_{in})}{\pi f T_{in}} \right] & \text{for } m \text{ even} \\ 2(-1)^{(m+1)/2} T_{in} m! \sin(2\pi f(n+\frac{1}{2})T_{in}) \Phi(m, f) & \text{for } m \text{ odd,} \end{cases} \tag{17}$$

where

$$\Phi(m, f) = \sum_{k=0}^{\lfloor (m-1)/2 \rfloor} (\pi f T_{in})^{2k-m} \frac{(-1)^k}{(2k)!} \left(\frac{\sin(\pi f T_{in})}{\pi f T_{in}} - \frac{\cos(\pi f T_{in})}{(2k+1)} \right). \tag{18}$$

It has turned out that for low values of f there are numerical accuracy problems because the summation of some relatively large consecutive terms in equation (18) is very small. This problem can be avoided by using power-series expansions for sine and cosine functions with low values of f, resulting in the form

$$\Phi(m, f) = \sum_{k=0}^{\lfloor (m-1)/2 \rfloor} \sum_{l=\lfloor (m-1)/2 \rfloor + 1}^{\infty} (\pi f T_{in})^{2(k+l)-m} \frac{(-1)^{k+l} 2(k-l)}{(2k+1)!(2l+1)!}. \tag{19}$$

It has been experimentally observed that a proper borderline for using equations (18) or (19) is at $f = 2.5F_{in}$. Furthermore, it has turned out that a good selection for the upper limit in the second summation of equation (19) is $\lfloor (m-1)/2 \rfloor + 20$.

The preceding form of the Fourier transform of $h_a(t)$ is very attractive because it is linear with respect to the unknowns $c_m(n)$. This enables one to optimize the unknowns very quickly in a manner similar to that of the linear-phase FIR filters, as will be seen in the next section. An alternative form for expressing $H_a(f)$ by using certain weighting functions has been proposed in [15].

[5] We point out that the derivation of the Fourier transform of the basis function $g(n,m,t)$ has been performed for the case where there are no restrictions at the above time instants due to the fact that these restrictions have no effect on the resulting Fourier transform. All that matters is that after including the time domain constraints, some $c_m(n)$'s are related to each other, resulting in some limitations on the frequency domain behavior of the interpolation filter, as will be seen in Section 5.

4.3. Modified Farrow structure

By substituting equation (12) into equation (4), it can be shown that the generalized interpolation filter can be implemented using a modified version of the original Farrow structure. This modified structure has two differences compared to the original structure. First, the output samples $v_m(n_l)$ of the FIR filters are multiplied by $2\mu_l - 1$, instead of μ_l. Second, the impulse responses of these $M + 1$ FIR filters with transfer functions of the form

$$C_m(z) = \sum_{k=0}^{N-1} c_m(k - N/2)z^{-k} \text{ for } m = 0, 1, \ldots, M \tag{20}$$

possess the properties given by equation (13). This implies that the impulse response coefficients $c_m(k)$ satisfy for $k = 0, 1, \ldots, N/2 - 1$

$$c_m(N/2 - 1 - k) = \begin{cases} c_m(-N/2 + k) & \text{for } m \text{ even} \\ -c_m(-N/2 + k) & \text{for } m \text{ odd.} \end{cases} \tag{21}$$

When exploiting the above symmetries, the number of coefficients to be implemented can be reduced from $(M + 1)N$ to $(M + 1)N/2$. In the original structure, such symmetries do not exist if the fractional interval is given in the interval $0 \le \mu_l < 1$.

The coefficients of the modified structure denoted by $c_m(n)$ and the coefficients of the original structure denoted by $\hat{c}_m(n)$ are related to each other via [19]

$$c_m(n) = \sum_{k=m}^{M} \frac{1}{2^k} \begin{pmatrix} k \\ m \end{pmatrix} \hat{c}_k(n), \tag{22}$$

for $n = -N/2, -N/2 + 1, \ldots, N/2 - 1$ and $m = 0, 1, \ldots, M$.

5. Filter optimization

This section shows how the proposed generalized interpolation filter can be optimized in the frequency domain in the minimax or in the least-mean-square sense subject to the given time domain conditions. We start by stating the constrained optimization problems and then convert these problems into the corresponding unconstrained ones. Finally, efficient algorithms are described for finding the optimum solution.

5.1. Optimization problems

This contribution concentrates on the following two optimization problems:

Minimax optimization problem

Given N, M, and a compact subset $X \subset [0, \infty)$[6] as well as a desired function $D(f)$ that is continuous for $f \in X$ and a weight function $W(f)$ that is positive for $f \in X$, find the $(M+1)N/2$ unknown coefficients $c_m(n)$ to minimize

$$\delta_\infty = \max_{f \in X} \left| W(f) \left[H_a(f) - D(f) \right] \right| \tag{23}$$

subject to the given time domain conditions of $h_a(t)$. Here, the frequency response $H_a(f)$ is given by equation (16).

Least-mean-square optimization problem.

Given the same parameters and functions as for the preceding problem, find the $(M+1)N/2$ unknown coefficients $c_m(n)$ to minimize

$$\delta_2 = \int_X \left\{ W(f) \left[H_a(f) - D(f) \right] \right\}^2 df \tag{24}$$

subject to the given time domain conditions of $h_a(t)$.

Any conditions that can be expressed in the form

$$\sum_{n=0}^{N/2-1} \sum_{m=0}^{M} a_{m,k}(n) c_m(n) \le b_k \quad \text{for } k = 1, 2, \ldots, K, \tag{25}$$

where the $a_{m,k}(n)$'s and b_k's are constants can be included in a straightforward manner in the preceding optimization problems. Here K is the number of conditions. Therefore, it is also possible to set some frequency domain conditions for $H_a(f)$. For instance, it may be desired that $H_a(f) = 1$[7] or 0 at some frequency point in the passband or stopband, respectively.

In most applications, there is no need to set any conditions for $h_a(t)$. In these cases, all degrees of freedom can be utilized for the frequency domain optimization. There are, however, applications where it is useful to give some time domain conditions. For instance, if the interpolation filter is used for approximating the derivative of $y_a(t)$, it may be desired that the first derivative of $h_a(t)$ be continuous [16]. In the sequel, the following four cases will be considered.

Case I: There are no time domain conditions.
Case II: $h_a(t)$ is continuous at $t = kT_{in}$ for $k = \pm 1, \pm 2, \ldots, \pm(N/2 - 1)$.
Case III: $h_a(0) = 1$ and $h_a(kT_{in}) = 0$ for $k = \pm 1, \pm 2, \ldots, \pm N/2$.
Case IV: The first derivative of $h_a(t)$ is continuous at $t = kT_{in}$ for $k = 0, \pm 1, \pm 2, \ldots, \pm(N/2 - 1)$.

[6] In most cases, X consists of disjoint passband and stopband regions, as will be seen later on.

[7] In the practical optimization, it is assumed that $F_{in} = 1$. For other values of F_{in}, $H_a(f)$ approximates $1/F_{in}$ in the passband. The above assumption is also true when plotting the magnitude responses in Section 6.

Case I is the most natural selection in many applications where the main goal is to provide the desired frequency selectivity for the reconstruction filter. Case II guarantees that the impulse response of the reconstruction filter is continuous. However, in the case of selective reconstruction filters, Case I and Case II designs are practically the same, indicating that the Case II conditions are not very restrictive. Case III is of importance if the original input sample values are required to be the same after interpolation. Case IV mainly applies when the interpolation filter is used for approximating the derivative of $y_a(t)$ [16]. Finally, we point out that various other time domain constraints may be included in a similar manner in the above optimization problems depending on the application.

5.2. Modified optimization problems

All of the preceding four cases are of equal type. Therefore, the time domain conditions can be easily dropped out and the optimization problems can be restated in the forms of unconstrained problems by properly changing the approximating function and the desired function. In Case I, there are no time domain conditions. Appendix B shows that in Cases II, III, and IV the constraints are met by relating, respectively, the $c_0(n)$s for $n = 1, 2, \ldots, N/2 - 1$, the $c_0(n)$s and $c_1(n)$s for $n = 0, 1, \ldots, N/2 - 1$, and the $c_1(n)$s for $n = 0, 1, \ldots, N/2 - 1$ to the remaining $c_m(n)$s according to Table 1.

The coefficients given in Table 1 are not included in the optimization because they can be calculated by using the remaining unknown coefficients.

By substituting the coefficients of Table 1 into equation (16), the approximating function $H_a(f)$ can be expressed as follows (see Appendix B):

$$H_a(f) = \tilde{H}_a(f) + E(f), \tag{26}$$

where $E(f)$ is a function that is independent of the unknowns, and $\tilde{H}_a(f)$ is a modified approximating function containing the remaining unknowns and is of the following form:

$$\tilde{H}_a(f) = \sum_{r=0}^{R-1} b(r)\Psi(r, f). \tag{27}$$

Here, the $\Psi(r, f)$'s are the modified basis functions and R is the number of remaining unknowns $b(r)$. Tables 1 through 3 give for each case the number of unknowns R, the function $E(f)$, and the relation between the coefficients $b(r)$ and $c_m(n)$ as well as that between the functions $\Psi(r, f)$ and $G(n, m, f)$.

Using equation (26) the weighted error function in equations (23) and (24) can be expressed as

$$W(f)\left[H_a(f) - D(f)\right] = W(f)\left[\tilde{H}_a(f) - \tilde{D}(f)\right], \tag{28}$$

where the new desired function is given by

$$\tilde{D}(f) = D(f) - E(f). \tag{29}$$

Table 1. Relations between coefficients $c_m(n)$ to meet Case I, II, III, and IV time domain conditions and the corresponding relations between modified and original basis functions $\Psi(r, f)$ for $r = 0, 1, \ldots, R-1$ and $G(n, m, f)$ for $n = 0, 1, \ldots, N/2-1$ and for $m = 0, 1, \ldots, M$[1]

CASE I	No conditions	$\Psi(n(M+1)+m, f) = G(n, m, f)$
CASE II	$c_0(1) = c_0(0) - \sum_{m=1}^{M} ((-1)^m c_m(1) - c_m(0))$ $c_0(n) = c_0(0) - \sum_{m=1}^{M} ((-1)^m c_m(n) - c_m(0))$ $+2 \sum_{\substack{m=1 \\ m \text{ odd}}}^{M} \sum_{k=1}^{n-1} c_m(k)$ for $n = 2, 3, \ldots, N/2-1$	$\Psi(m, f) = G(0, m, f) + \sum_{k=1}^{N/2-1} G(k, 0, f)$ $\Psi(nM+m, f) = G(n, m, f) - G(n, 0, f)$ for $n > 0$ and $m > 0$ even $\Psi(nM+m, f) = G(n, m, f) + G(n, 0, f)$ $+2 \sum_{k=n+1}^{N/2-1} G(k, 0, f)$ for $n > 0$ and m odd
CASE III	$c_0(0) = 1/2 - \sum_{\substack{m=2 \\ m \text{ even}}}^{M} c_m(0), c_1(0) = -1/2 - \sum_{\substack{m=3 \\ m \text{ odd}}}^{M} c_m(0)$ $c_0(n) = - \sum_{\substack{m=2 \\ m \text{ even}}}^{M} c_m(n), \quad c_1(n) = - \sum_{\substack{m=3 \\ m \text{ odd}}}^{M} c_m(n)$ for $n = 1, 2, \ldots, N/2-1$	$\Psi(n(M-1)+m-2, f) = G(n, m, f) - G(n, \alpha, f)$, where $\alpha = \begin{cases} 0 \text{ for } m \text{ even and } m > 0 \\ 1 \text{ for } m \text{ odd and } m > 1 \end{cases}$
CASE IV	$c_1(0) = \sum_{m=2}^{M} (-1)^m m c_m(0)$ $c_1(n) = 2 \sum_{k=0}^{n-1} \left[\sum_{\substack{m=2 \\ m \text{ even}}}^{M} m c_m(k) + \sum_{m=2}^{M} (-1)^m m c_m(n) \right]$ for $n = 1, 2, \cdots, N/2-1$	$\Psi(nM, f) = G(n,0,f)$ $\Psi(nM+m-1, f) = G(n, m, f) + mG(n, 1, f)$ $+2m \sum_{k=n+1}^{N/2-1} G(k, 1, f)$ for m even and $m > 0$ $\Psi(nM+m-1, f) = G(n, m, f) - mG(n, 1, f)$ for m odd and $m > 1$

[1] Note that the indices n and m corresponding to the drop-out $c_m(n)$s are disregarded when forming the relations between the $\Psi(r, f)$s and $G(n, m, f)$s

Table 2. Number of unknown coefficients R and functions $E(f)$ for the four different cases

	R	$E(f)$
CASE I	$N(M+1)/2$	0
CASE II	$NM/2+1$	0
CASE III	$N(M-1)/2$	$(G(0,0,f)-G(0,1,f))/2$
CASE IV	$NM/2$	0

Table 3. Relations between the modified unknowns $b(r)$ for $r = 0, 1, \ldots, R-1$ and the original unknowns $c_m(n)$ for $n = 0, 1, \ldots, N/2-1$ and for $m = 0, 1, \ldots, M$. Note that the indices n and m corresponding to the drop-out $c_m(n)$s, as indicated in Table 1, are disregarded in these relations

CASE I	$b(n(M+1)+m) = c_m(n)$ for $m = 0, 1, \ldots, M$
CASE II	$b(0) = c_0(0)$
	$b(nM+m) = c_m(n)$ for $m = 1, 2, \ldots, M$
CASE III	$b(n(M-1)+m-2) = c_m(n)$ for $m = 2, 3, \ldots, M$
CASE IV	$b(nM) = c_0(n)$
	$b(nM+m-1) = c_m(n)$ for $m = 2, 3, \ldots, M$

Based on these equations, the original constrained problems can be transformed in all the four cases into the following unconstrained ones:

Modified unconstrained minimax optimization problem.

Given the same parameters and functions as for the original problem, find the R unknown coefficients $b(r)$ to minimize

$$\delta_\infty = \max_{f \in X} \left| W(f) \left[\tilde{H}_a(f) - \tilde{D}(f) \right] \right|, \tag{30}$$

where $\tilde{H}_a(f)$ is given by equation (27) and $\tilde{D}(f)$ is given by equation (29).

Modified unconstrained least-mean-square optimization problem.

Find the R unknown coefficients $b(r)$ to minimize

$$\delta_2 = \int_X \left\{ W(f) \left[\tilde{H}_a(f) - \tilde{D}(f) \right] \right\}^2 df. \tag{31}$$

There are two main advantages of using these problems. First, the unconstrained problems are more straightforward to solve, as will be seen in the following two subsections. Second, finding the optimum solution is significantly faster. After finding the unknowns $b(r)$ for $r = 0, 1, \ldots, R-1$, the corresponding coefficients $c_m(n)$ can be determined from Table 3 and, then, the remaining coefficients can be solved according to the equations of Table 1.

5.3. Optimization in the minimax sense

The minimization of δ_∞, as given by equation (30), can be performed conveniently by using linear programming. This can be carried out by using the following two steps (see, e.g., [13]).

(1) Sample $\tilde{D}(f)$ and $W(f)$ as well as the $\Psi(r, f)$'s along a dense grid of frequencies $f_1, f_2, \ldots, f_K$ on X.

(2) Apply linear programming to find $b(r)$ for $r = 0, 1, \ldots, R-1$ and δ_∞ subject to the constraints

$$\sum_{r=0}^{R-1} b(r)\Psi(r, f_k) - \frac{\delta_\infty}{W(f_k)} \le \tilde{D}(f_k) \tag{32a}$$

and

$$-\sum_{r=0}^{R-1} b(r)\Psi(r, f_k) - \frac{\delta_\infty}{W(f_k)} \le -\tilde{D}(f_k) \tag{32b}$$

for $k = 1, 2, \ldots, K$ such that δ_∞ is minimized. Note that the maximum deviation δ_∞ is also an unknown.

5.4. Optimization in the least-mean-square sense

In this case, $\tilde{D}(f)$ and $W(f)$ as well as the $\Psi(r,f)$'s are sampled along a dense grid of frequencies $f_1, f_2, \ldots, f_K$ on X, and δ_2 as given by equation (31) can be replaced by the following summation:

$$\delta_2 \approx \hat{\delta}_2 = \frac{1}{K}\sum_{k=1}^{K}\left(W(f_k)\left(\sum_{r=0}^{R-1} b(r)\Psi(r, f_k) - \tilde{D}(f_k)\right)\right)^2. \tag{33}$$

Before giving the optimum solution, the following matrix and two vectors are constructed:

$$\mathbf{X} = \begin{bmatrix} W(f_1)\Psi(0, f_1) & W(f_1)\Psi(1, f_1) & \cdots & W(f_1)\Psi(R-1, f_1) \\ W(f_2)\Psi(0, f_2) & W(f_2)\Psi(1, f_2) & \cdots & W(f_2)\Psi(R-1, f_2) \\ \vdots & \vdots & & \vdots \\ W(f_K)\Psi(0, f_K) & W(f_K)\Psi(1, f_K) & \cdots & W(f_K)\Psi(R-1, f_K) \end{bmatrix}, \tag{34a}$$

$$\mathbf{b} = \begin{bmatrix} b(0) & b(1) & \cdots & b(R-1) \end{bmatrix}^T, \tag{34b}$$

and

$$\mathbf{d} = \left[W(f_1)\tilde{D}(f_1) \quad W(f_2)\tilde{D}(f_2) \quad \cdots \quad W(f_K)\tilde{D}(f_K) \right]^T . \tag{34c}$$

Here, **b** is the vector containing the unknowns. It is well known that **b** minimizing $\hat{\delta}_2$, as given by equation (33), is [10]

$$\mathbf{b} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1} \mathbf{X}^T\mathbf{d} \tag{35a}$$

and it satisfies the "normal equations"

$$\left(\mathbf{X}^T\mathbf{X}\right)\mathbf{b} = \mathbf{X}^T\mathbf{d}. \tag{35b}$$

Equation (33) is an approximation to the least-mean-square error as given by equation (31). The approximation accuracy can be improved by increasing the number of frequency points denoted by K. In practice, K should be limited in order to have a reasonable computational complexity, and Gaussian elimination or some other techniques should be used in equation (35a) instead of matrix inversion (see, e.g., [10]). Furthermore, a good accuracy for the solution obtained by minimizing $\hat{\delta}_2$, as given by equation (33), implies that the grid points are selected equidistantly. In most problems the upper limit for X is infinity. In practice, it has turned out that a proper selection for the upper limit is between $30F_{in}$ and $100F_{in}$, depending on the length and polynomial degree of the interpolation filter. For K, a proper selection is between 10^3 and 10^4.

6. Examples

This section provides two design examples to illustrate the flexibility of the proposed minimax and least-mean-square synthesis methods as well as the performance of the resulting filters. These filters are compared to those obtained using the conventional design methods.

Example 1. In this example, the highest baseband frequency component of the input signal $x(n)$ is assumed to be at $0.35F_{in}$, and it is desired that the polynomial-based interpolation filter preserve the baseband signal with at most 1% error in the linear scale and attenuate the image frequencies by at least 60 dB. Hence, the filter specifications are as follows. The passband edge is located at f_p=$0.35F_{in}$, the stopband region consists of the intervals $[(k - 0.35)F_{in}, (k + 0.35)F_{in}]$ for $k = 1, 2, \ldots$, the maximum passband deviation of the magnitude response from unity is $\delta_p = 0.01$, and the minimum stopband attenuation is $A_s = 60$ dB in the stopband region.

Altogether, five design methods are under consideration, namely, the Lagrange and B-spline interpolations [5], [14], the least-mean-square synthesis technique of Farrow that is based on the FD filter approach [6] (referred to as the "L_2 *Farrow*" technique) as well as the least-mean-square (referred to as "L_2" technique) and the minimax optimization techniques proposed in this contribution. In addition, two alternatives are considered for each of these methods. The first one is the direct implementation using the modified Farrow structure. A cascade of the prefilter and the modified Farrow structure is assumed to be used for the B-spline interpolation, where the prefilter is approximated by a linear-phase FIR filter of a proper order, see, e.g., [4]. For the second alternative, the sampling rate is increased by a factor of two with the aid of a fixed linear-phase FIR interpolator before using the Farrow structure. This FIR filter is optimized separately for each case using the Remez algorithm. The overall system comprising the fixed FIR filter and the Farrow structure has to meet the filter specifications.

Table 4 shows data for each of the five design methods in the above-mentioned two alternative cases. In this table, L is the interpolation factor of the fixed digital interpolator. For the second alternative $L = 2$, whereas for the first alternative $L = 1$ and the fixed interpolator is not in use. According to the previous discussions, N and M indicate that the modified Farrow structure consists of $M + 1$ FIR filters of length N with either a symmetrical or anti-symmetrical impulse response. δ_p is the maximum passband deviation from unity in the linear scale, and A_s is the minimum stopband attenuation. For $L = 2$, N_{FIR}, the length of the FIR filter used for the fixed interpolation, is given. In addition, the number of multipliers required in the overall implementation, including the multiplications by μ_l, and the multiplication rate, that is, the number of multiplications per input sample, are included in the table. In determining these quantities, the coefficient symmetries of the modified Farrow structure and the FIR filter as well as the interpolation ratio $\beta = F_{out}/F_{in}$ are taken into account.

As seen from Table 4, the conventional time domain methods, namely, the Lagrange and B-spline interpolations, have clearly higher implementation complexities, in terms of the number of multipliers and the multiplication rate, when compared with the three remaining methods. As can be expected, the minimax filter with $L = 2$ provides the lowest complexity. In addition, the number of multipliers can be reduced for all five cases by using a fixed linear-phase FIR interpolator with $L = 2$. The multiplication rate is also much smaller for the $L = 2$ case if the overall sampling rate conversion factor β is large enough. Note that in the $L = 1$ case, the first three filters have a very low passband deviation because these methods do not allow the use of different weights for shaping the passband and the multiband stopband region.

The Minimax I filter in Table 4 has been optimized using an iterative algorithm, where, in each iteration, the FIR interpolator or the Farrow structure is optimized in the minimax sense using linear programming so that they properly share the frequency-response-shaping responsibilities and the effect of the magnitude response of the other filter is properly included in the desired and weighting functions. The advantage of this joint optimization becomes more evident in the next example.

As examples, Figures 8a and b show the magnitude responses of the Lagrange

Table 4. Example 1 with nonuniform stopband

	Interpolation factor L	N	M	δ_p	A_s (dB)	N_{FIR}	Number of multipliers	Multiplication rate
Lagrange	1	42	41	0.001	61.8	–	923	882+41β
B-spline [(1)]	1	12	11	0.0006	64.5	–	89	78+11β
L_2 Farrow	1	12	4	0.0008	63.3	–	34	30+4β
L_2	1	12	4	0.01	64.7	–	34	30+4β
Minimax	1	10	4	0.01	67.3	–	29	25+4β
Lagrange	2	8	7	0.01	65.8	20	49	74+7β
B-spline [(1]	2	6	5	0.01	65.9	20	39	58+5β
L_2 Farrow	2	6	3	0.01	66.8	20	25	34+3β
L_2	2	6	3	0.01	66.5	20	25	34+3β
Minimax	2	4	3	0.01	62.2	20	21	26+3β
Minimax I [(2)]	2	4	3	0.01	64.8	19	21	26+3β

[(1)] The length of the prefilter is 11. When exploiting the coefficient symmetries, six multipliers are required.

[(2)] The Farrow structure and the FIR filter are jointly optimized to meet the specifications.

and the B-spline interpolation filters for several values of M, whereas Figure 8c shows the magnitude response of the Farrow interpolation filter [6] for the following design parameters: the length of the filter is $N = 12$, the degree of the polynomials is $M = 4$, and the passband edge is located at $f_p = 0.35F_{in}$.

Figure 9 shows the magnitude responses of the minimax filter for $L = 1$ and for $L = 2$ in the case where the fixed interpolator and the Farrow structure have been separately designed such that the overall filter and the subfilters have the same bassband region.

Example 2. The filter specifications in this example are as follows: $f_p = 0.4F_{in}$, $f_s = 0.6F_{in}$, $\delta_p = 0.001$, and $A_s = 80$ dB. The main difference compared to the previous example is that the stopband region is uniform starting at $0.6F_{in}$. A uniform stopband is needed if there are some undesired signal components, e.g., wideband noise in the frequencies between the desired signal components and the aliasing components.

Various fixed interpolation ratios have been used for the five design methods summarized in Table 5. For the interpolation factors $L = 4$ and $L = 6$, a two-stage fixed linear-phase FIR interpolator of lengths N_{FIR1} and N_{FIR2} has been designed so that the interpolation ratio for the first stage is 2.

The problem with the conventional design methods (Lagrange, B-spline, and L_2 Farrow) is that they do not utilize the analog model and, therefore, "don't care" bands always exist between the image frequencies. Consequently, in the direct implementation ($L = 1$) these filters do not provide a uniform stopband region with reasonably low values of N and M.

As seen from Table 5, the interpolation filters obtained with the joint optimization (filters Minimax I-III) have much lower complexities in terms of the number of multipliers and the multiplication rate. Figure 10 shows the magnitude responses for the L_2 and Minimax interpolation filters in the case of direct implementation ($L = 1$) as well as for the Minimax and Minimax II filters with $L = 4$. As seen from Figure 10d, the joint optimization of the two stage-interpolator with both interpolator factors equal to two and the Farrow structure has been performed as follows. The first interpolation stage concentrates on shaping the overall response in the region $[0, F_{in}]$. The role of the second interpolation stage and the Farrow structure is then to properly attenuate the extra unwanted passbands and transition bands of the first interpolation stage so that the magnitude responses achieve a value of unity at the passband edge $f_p = 0.4F_{in}$.

7. Conclusions

The main contribution of this paper was a proposed frequency domain optimization method for a generalized class of polynomial-based interpolation filters. This class of interpolation filters was derived by using the analog model for the general interpolation problem that enables one to exploit the frequency domain information of the input signal when optimizing the interpolation filters. It was shown that

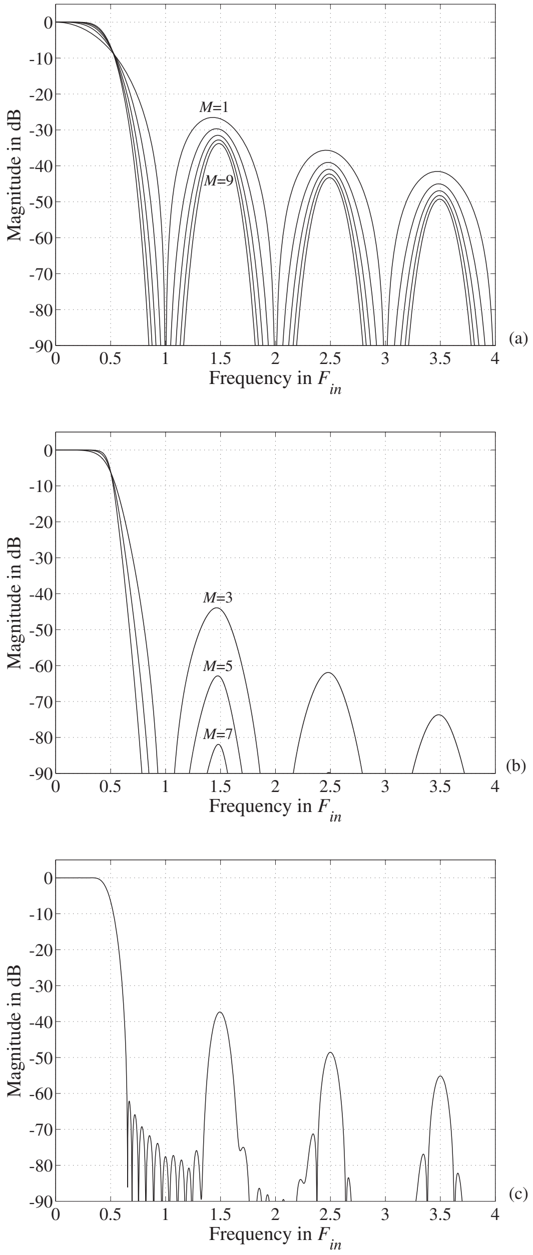
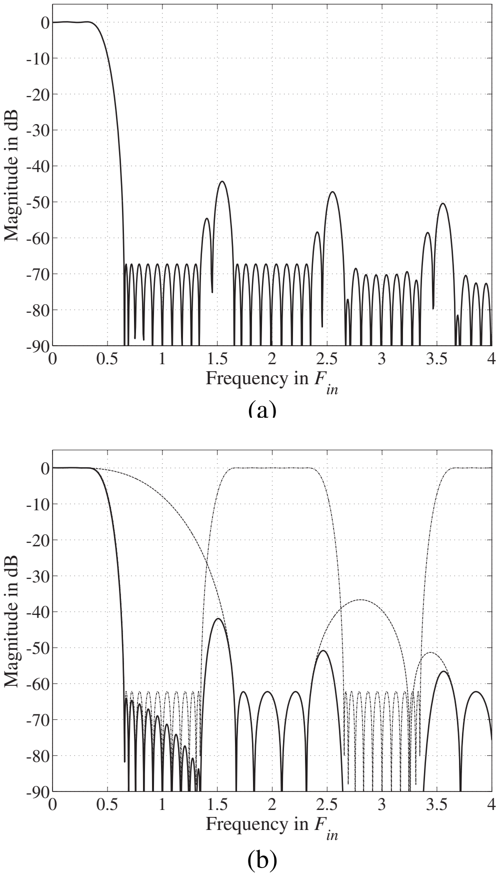


Figure 8. The magnitude responses for (a) the Lagrange interpolation filters with degrees $M = 1, 3, 5, 7$, and 9, (b) the B-spline interpolation filters [6] with degrees $M = 3, 5$, and 7, and (c) the Farrow interpolation filter for $N = 12$, $M = 4$, and $f_p = 0.35F_{in}$.



(a)

(b)

Figure 9. The overall magnitude responses of the minimax interpolation filters of Example 1 (solid lines) for (a) $L = 1$ and (b) $L = 2$. For $L = 2$, the Farrow structure and the fixed FIR filter have been separately designed and their magnitude responses are given by the dashed and dash-dotted lines, respectively.

the coefficients of the reconstruction filter in the analog model uniquely determine the coefficients for the modified Farrow structure.

The minimax and least-mean-square optimizations of the filter coefficients were given. The parameters for the optimization are the length of the filter, the degree of the interpolation, the passband and stopband regions, the desired response, and the weighting function.

Examples indicated that the proposed optimizations offer filters with lower complexities compared with the existing interpolation filters, especially when the fixed FIR interpolator and the Farrow structure are jointly optimized to meet the filter specifications. A set of Matlab codes for proposed interpolation filters are available via http://www.cs.tut.fi/ ts.

Table 5. Example 2 with uniform stopband

	Fixed Int. Factor L	N	M	δ_p	A_s (dB)	N_{FIR1}	N_{FIR2}	Number of multipliers	Multiplication rate
Lagrange [(1)]	1	-	-	-	-	-	-	-	-
B-spline [(2)]	1	24	23	0.0007	85.0	-	-	317	294+23β
L_2 Farrow [(1)]	1	-	-	-	-	-	-	-	-
L_2	1	24	7	0.001	81.1	-	-	103	96+7β
Minimax	1	22	5	0.001	87.4	-	-	71	66+5β
Lagrange	2	18	17	0.001	82.8	41	-	200	345+17β
B-spline [(2)]	2	8	7	0.001	82.8	41	-	66	97+7β
L_2 Farrow	2	10	4	0.001	82.5	41	-	50	71+4β
L_2	2	8	4	0.001	80.1	41	-	45	61+4β
Minimax	2	8	4	0.001	82.1	41	-	45	61+4β
Lagrange	4	8	7	0.001	84.0	43	17	70	168+7β
B-spline [(2)]	4	6	5	0.001	82.7	43	17	60	136+5β
L_2 Farrow	4	6	3	0.001	84.2	43	17	46	88+3β
L_2	4	6	3	0.001	84.3	43	17	46	88+3β
Minimax	4	6	3	0.001	84.3	43	17	46	88+3β
Minimax I [(3)]	2	6	4	0.001	84.6	39	-	39	50+4β
Minimax II [(3)]	4	4	3	0.001	83.5	40	7	35	60+3β
Minimax III [(3)]	6	4	2	0.001	80.0	40	11	34	68+2β

[(1)] There are always high ripple values between the image frequencies. Therefore, a uniform stopband would require extremely high values for N and M.

[(2)] The length of the prefilter is 11.

[(3)] The Farrow structure and the FIR filter(s) are jointly optimized to meet the specifications.

Appendix A

This appendix shows that the Fourier transform of the basis function $g(n, m, t)$ as given by equation (14) can be expressed as given by equations (17) and (18). This transform, denoted by $G(n, m, f)$, is real valued because $g(n, m, t)$ is symmetrical around $t = 0$ for all values of n and m (except at the points kT_{in} for $k = \pm 1, \pm 2, \ldots, \pm N/2$.

To simplify the derivation, $g(n, m, t)$ is rewritten in the following form:

$$g(n, m, t) = (-1)^m \hat{f}(m, t + (n + 1/2)T_{in}) + \hat{f}(m, t - (n + 1/2)T_{in}), \quad \text{(A1)}$$

where

$$\hat{f}(m, t) = \left(\frac{2t}{T_{in}}\right)^m (u(t + T_{in}/2) - u(t - T_{in}/2)), \quad \text{(A2)}$$

where $u(t)$ is the unit step function. Alternatively, $\hat{f}(m, t)$ can be expressed as

$$\hat{f}(m, t) = e_1(m, t + T_{in}/2) - e_2(m, t - T_{in}/2), \quad \text{(A3)}$$

where

$$e_1(m, t) = \left(\frac{2t}{T_{in}} - 1\right)^m u(t) = \sum_{k=0}^{m} (-1)^k \binom{m}{k} \left(\frac{2t}{T_{in}}\right)^{m-k} u(t), \quad \text{(A4)}$$

and

$$e_2(m, t) = \left(\frac{2t}{T_{in}} + 1\right)^m u(t) = \sum_{k=0}^{m} \binom{m}{k} \left(\frac{2t}{T_{in}}\right)^{m-k} u(t). \quad \text{(A5)}$$

Based on the following two facts,

(1) if the Laplace transform of $c(t)$ is $C(s)$, then the Laplace transform of $c(t - K)$ is $e^{-sK}C(s)$, and
(2) the Laplace transform of $t^m u(t)$ is $m!/s^{m+1}$,

the Laplace transform of $g(n, m, t)$ takes the following form,

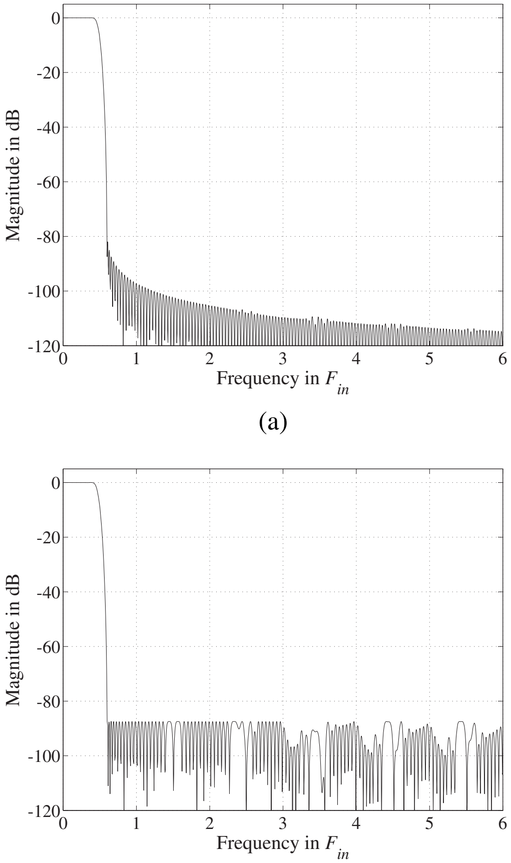
$$G(n, m, s) = (-1)^m e^{s(n+\frac{1}{2})T_{in}} \hat{F}(m, s) + e^{-s(n+\frac{1}{2})T_{in}} \hat{F}(m, s), \quad \text{(A6)}$$

where

$$\hat{F}(m, s) = e^{sT_{in}/2} E_1(m, s) - e^{-sT_{in}/2} E_2(m, s) \quad \text{(A7)}$$

with

$$\begin{aligned} E_1(m, s) &= \sum_{k=0}^{m} (-1)^k \binom{m}{k} \left(\frac{2}{T_{in}}\right)^{m-k} \frac{(m-k)!}{s^{m-k+1}} \\ &= \sum_{k=0}^{m} (-1)^k \left(\frac{2}{T_{in}}\right)^{m-k} \frac{m!}{k! s^{m-k+1}} \end{aligned} \quad \text{(A8)}$$



(a)

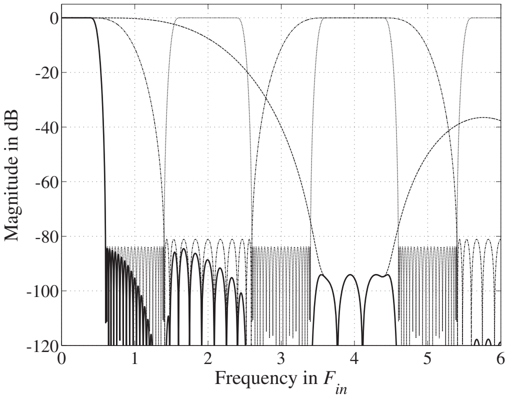
(b)

Figure 10. Magnitude responses for interpolation filters in Example 2 included in Table 5. (a) L_2 interpolation filter for $L = 1$ (b) Minimax interpolation filter for $L = 1$. (*Cont.*).

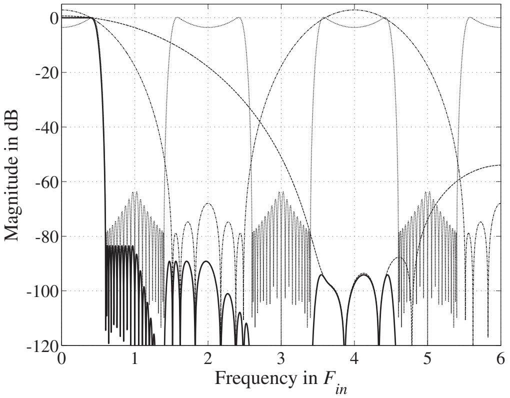
and

$$E_2(m, s) = \sum_{k=0}^{m} \left(\begin{array}{c} m \\ k \end{array} \right) \left(\frac{2}{T_{in}} \right)^{m-k} \frac{(m-k)!}{s^{m-k+1}} = \sum_{k=0}^{m} \left(\frac{2}{T_{in}} \right)^{m-k} \frac{m!}{k! s^{m-k+1}}. \tag{A9}$$

The Fourier transform of the basis function $g(n, m, t)$ is now obtained by using



(c)



(d)

Figure 10. (*Cont.*) (c) Minimax filter for $L = 4$. The solid, dashed, dotted, and dash-dotted lines show the responses for the overall filter, the Farrow structure, the first fixed interpolation filter, and the second interpolator filter, respectively. (d) The same as in (c) for the jointly optimized Minimax II filter.

the substitution $s = j2\pi f$, that is,

$$\begin{aligned} G(n, m, f) &\equiv G(n, m, j2\pi f) \\ &= \left((-1)^m e^{j2\pi f(n+\frac{1}{2})T_{in}} + e^{-j2\pi f(n+\frac{1}{2})T_{in}}\right)\hat{F}(m, j2\pi f) \\ &= \begin{cases} 2\cos(2\pi f(n+\frac{1}{2})T_{in})\hat{F}(m, j2\pi f) & \text{for } m \text{ even} \\ -2j\sin(2\pi f(n+\frac{1}{2})T_{in})\hat{F}(m, j2\pi f) & \text{for } m \text{ odd.} \end{cases} \end{aligned} \tag{A10}$$

After some manipulations, including normalization with $1/T_{in}$, this $G(n, m, f)$ can be expressed in the form given by equation (17).

Appendix B

This appendix shows how the Case II, III, and IV time domain conditions of Section 5 can be dropped out by properly converting the original constrained optimization problems to the corresponding unconstrained ones. The following notation is used here to simplify the derivations:

$$h(n,\ \mu_l) \equiv h_a((n+\mu_l)T_{in}), \tag{B1}$$

where the polynomial-based impulse response $h_a((n+\mu_l)T_{in})$ is given by equation (12).

B.1. Case II

In this case, the condition that $h_a(t)$ is continuous at $t = kT_{in}$ for $k = \pm 1, \pm 2, \ldots, \pm(N/2-1)$ can be stated in terms of the $h(n,\ \mu_l)$'s as follows:

$$h(n-1, 1) = h(n, 0) \tag{B2}$$

for $n = 1, 2, \ldots, N/2-1$. By using equations (B1) and (12), the condition of equation (B2) can be given by means of the coefficients $c_m(n)$ as follows:

$$\sum_{m=0}^{M} c_m(n-1) = \sum_{m=0}^{M} (-1)^m c_m(n) \tag{B3}$$

for $n = 1, 2, \ldots, N/2-1$. These conditions are satisfied by relating the $c_0(n)$'s for $n = 1, 2, \ldots, N/2-1$ to the remaining $c_m(n)$'s according to Table 1. Substituting these $c_0(n)$'s into equation (16) gives

$$H_a(f) = \tilde{H}_a(f) + E(f), \tag{B4}$$

where

$$E(f) = 0 \tag{B5}$$

and

$$\tilde{H}_a(f) = c_0(0)\tilde{G}(0, 0, f) + \sum_{n=0}^{N/2-1} \sum_{m=1}^{M} c_m(n)\tilde{G}(n, m, f) \tag{B6}$$

with

$$\tilde{G}(n,m,f) = \begin{cases} G(0,m,f) + \sum_{k=1}^{N/2-1} G(k,0,f) & \text{for } n = 0 \\ G(n,m,f) - G(n,0,f) & \text{for } n > 0 \text{ and } m \text{ even} \\ G(n,m,f) + G(n,0,f) & \\ \quad +2 \sum_{k=n+1}^{N/2-1} G(k,0,f) & \text{for } n > 0 \text{ and } m \text{ odd.} \end{cases} \tag{B7}$$

Here, $G(n, m, f)$ is given by equations (17) and (18) and, for $f \geq 1.5F_{in}$ and $f < 1.5F_{in}$, by equations (17) and (19) respectively.

Based on the above derivation, the modified unconstrained optimization problems take the forms of equations (30) and (31), and the number of unknowns R, functions $E(f)$, coefficients $b(r)$, and the modified basis functions $\Psi(r,f)$ become as given in Tables 1–3.

B.2. Case III

In this case, it is required that $h_a(0) = 1$ and $h_a(kT_{in}) = 0$ for $k = \pm 1, \pm 2, \ldots, \pm N/2$. These criteria can be stated in terms of the $h(n, \mu_l)$'s for $n = 0$ as

$$h(0,\ 0) = \sum_{m=0}^{M} (-1)^m c_m(0) = 1 \quad \text{and} \quad h(0,\ 1) = \sum_{m=0}^{M} c_m(0) = 0 \tag{B8}$$

and for $n = 1, 2, \ldots, N/2 - 1$ as

$$h(n,\ 0) = \sum_{m=0}^{M} (-1)^m c_m(n) = 0 \quad \text{and} \quad h(n,\ 1) = \sum_{m=0}^{M} c_m(n) = 0. \tag{B9}$$

These conditions are satisfied by relating the $c_0(n)$'s and $c_1(n)$'s for $n = 0, 1, \ldots, N/2-1$ to the remaining $c_m(n)$'s according to Table 1. Substituting these $c_0(n)$'s and $c_1(n)$'s into equation (16) gives

$$H_a(f) = \tilde{H}_a(f) + E(f), \tag{B10}$$

where

$$E(f) = (G(0, 0, f) - G(0, 1, f))/2 \tag{B11}$$

and

$$\tilde{H}_a(f) = \sum_{n=0}^{N/2-1} \sum_{m=2}^{M} c_m(n)\tilde{G}(n, m, f) \tag{B12}$$

with

$$\tilde{G}(n, m, f) = \begin{cases} G(n, m, f) - G(n, 0, f) & \text{for } m \text{ even} \\ G(n, m, f) - G(n, 1, f) & \text{for } m \text{ odd.} \end{cases} \tag{B13}$$

Tables 1–3 give the resulting R, $E(f)$, $b(r)$'s, and $\Psi(r, f)$'s for the corresponding modified unconstrained optimization problems.

B.3. Case IV

In this case, it is required that the first derivative of $h_a(t)$ be continuous at $t = kT_{in}$ for $k = 0$ and for $k = \pm 1, \pm 2, \ldots, \pm(N/2 - 1)$. These conditions can be stated by first taking the derivative of $h(n, \mu_l)$ as follows:

$$h'(n, \mu_l) = \frac{dh(n, \mu_l)}{d\mu_l} = \sum_{m=1}^{M} 2mc_m(n)(2\mu_l - 1)^{m-1}. \tag{B14}$$

It is required that

$$h'(0, 0) = 0 \tag{B15}$$

and for $n = 1, 2, \ldots, N/2 - 1$

$$h'(n - 1, 1) = h'(n, 0). \tag{B16}$$

The Case IV criteria are satisfied by relating the $c_1(n)$'s to the remaining $c_m(n)$'s for $n = 0, 1, \ldots, N/2 - 1$ according to Table 1. Substituting these $c_1(n)$'s into equation (16) gives

$$H_a(f) = \tilde{H}_a(f) + E(f), \tag{B17}$$

where

$$E(f) = 0 \tag{B18}$$

and

$$\tilde{H}_a(f) = \sum_{n=0}^{N/2-1} \sum_{\substack{m=0 \\ m \neq 1}}^{M} c_m(n)\tilde{G}(n, m, f) \tag{B19}$$

with

$$\tilde{G}(n, m, f) = \begin{cases} G(n, 0, f) & \text{for } m = 0 \\ G(n, m, f) + mG(n, 1, f) & \\ +2m \sum_{k=n+1}^{N/2-1} G(k, 1, f) & \text{for } m \text{ even} \\ G(n, m, f) - mG(n, 1, f) & \text{for } m \text{ odd.} \end{cases} \tag{B20}$$

Tables 1–3 give R, $E(f)$, $b(r)$, and $\Psi(r, f)$ for the corresponding modified unconstrained optimization problems.

References

[1] D. Babic, A. S. H. Ghadam, and M. Renfors, Polynomial-based filters with odd number of polynomial segments for interpolation, *IEEE Signal Processing Letters*, vol. 11, pp. 171–174, Feb. 2004.

[2] D. Babic, J. Vesma, T. Saramäki, and M. Renfors, Implementation of the transposed Farrow structure, in *Proc. IEEE Int. Symp. on Circuit and Systems*, Scottsdale, AZ, May 2002, pp. 5–8.

[3] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1983.

[4] K. Egiazarian, T. Saramäki, H. Chugurian, and J. Astola, Modified B-spline interpolators and filters: Synthesis and efficient implementation, in *Proc. Int. Conf. on Acoust., Speech, Signal Processing,* Atlanta, GA, May 1996, pp. 1743–1746.

[5] L. Erup, F. M. Gardner, and R. A. Harris, Interpolation in digital modems–Part II: Implementation and performance, *IEEE Trans. Commun.*, vol. 41, pp. 998–1008, June 1993.

[6] C. W. Farrow, A continuously variable digital delay element, in *Proc. IEEE Int. Symp. on Circuits and Systems*, Espoo, Finland, June 1988, pp. 2641–2645.

[7] F. M. Gardner, Interpolation in digital modems–Part I: Fundamentals, *IEEE Trans. Commun.*, vol. 41, pp. 501–507, Mar. 1993.

[8] f. harris, Performance and design considerations of Farrow filter used for arbitrary resampling, in *Proc. 13th Int. Conf. on Digital Signal Processing*, Santorini, Greece, July 1997, pp. 595–599.

[9] T. Hentschel and G. Fettweis, Continuous-time digital filters for sample rate conversion in reconfigurable radio terminals, in *Proc. of European Wireless 2000*, Sep. 12–14, Dresden, Germany, pp. 55–59.

[10] D. Kincaid and W. Cheney, *Numerical Analysis*, Brooks/Cole Publishing Company, 1996.

[11] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, Splitting the unit delay, *IEEE Signal Processing Magazine*, vol. 13, pp. 30–60, Jan. 1996.

[12] T. A. Ramstad, Digital methods for conversion between arbitrary sampling frequencies, *IEEE Trans. Acoust. Speech, Signal Processing*, vol. 32, pp. 577–591, June 1984.

[13] T. Saramäki, Finite impulse response filter design, Chapter 4 in *Handbook for Digital Signal Processing*, edited by S. K. Mitra and J. F. Kaiser, John Wiley & Sons, New York, 1993.

[14] M. Unser, A. Aldroubi, and M. Eden, Fast B-spline transforms for continuous image representation and interpolation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, pp. 277–285, March 1991.

[15] J. Vesma, A frequency-domain approach to polynomial-based interpolation and the Farrow structure, *IEEE Trans. on Circuits and Systems II*, vol. 47, pp. 206–209, March 2000.

[16] J. Vesma, *Optimization and applications of polynomial-based interpolation filters*, Dr. Thesis, Tampere, Finland: Tampere University of Tech., Department of Information Technology, May 1999.

[17] J. Vesma, R. Hamila, T. Saramäki, and M. Renfors, Design of polynomial interpolation filters based on Taylor series, in *Proc. IX European Signal Processing Conf.*, Rhodes, Greece, Sep. 1998, pp. 283–286.

[18] J. Vesma, F. Lopez, T. Saramäki, and M. Renfors, The effects of quantizing the fractional interval in interpolation filters, in *Proc. IEEE Nordic Signal Processing Symp.*, Kolmården, Sweden, June 2000, pp. 215–218.

[19] J. Vesma and T. Saramäki, Design and properties of polynomial-based fractional delay filters, in *Proc. IEEE Int. Symp. on Circuits and Systems*, Geneva, Switzerland, May 2000, pp. 104–107.

[20] J. Vesma and T. Saramäki, Interpolation filters with arbitrary frequency response for all-digital receivers, in *Proc. IEEE Int. Symp. on Circuits and Systems*, Atlanta, GA, May 1996, pp. 568–571.

[21] Li Wenzhen and M. Tomisawa, A modified transposed Farrow solution to multipurpose multirate filtering in software defined radio (SDR), in *Proc. IEEE Personal, Indoor and Mobile Radio Communications*, Sept. 2004, pp. 2725–2729.