International Journal of Computer Science and Applications

Vol. 10 No. 1, pp. 98 - 116, 2013

ON SAMPLE RATE CONVERSION BASED ON VARIABLE FRACTIONAL DELAY FILTERS

MAREK BLOK

Department of Teleinformation Networks
Faculty of Electronics, Telecommunications and Informatics
Gdańsk University of Technology
Narutowicza 11/12, 80-233 Gdańsk-Wrzeszcz, Poland
mblok@eti.pg.gda.pl

The sample rate conversion algorithm based on variable fractional delay filters is often used if the resampling ratio cannot be expressed as the ratio of small integer numbers or if it is not constant. The main advantage of such solution is that it allows for arbitrary resampling ratios which can even be changed during the resampling process. In this paper a discussion on influence of different approaches to fractional filter design on performance of sample rate conversion algorithm implemented using variable fractional delay filters is presented. Since the performance of such resampling algorithm depends solely on the method used to design fractional delay filters, we present its properties in relation to the presented classification of fractional delay filter design methods. The proposed general categories of fractional filter design are: optimal design, offset window method and polyphase decomposition.

Keywords: Sample rate conversion; variable fractional delay; fractional delay filter design.

1. Introduction

Digital representation of analog signals has a lot of advantages but the problem arise when the sample rate with which signal was recored is different from the sample rate required for further processing. With a large number of sample rate standards [AES (2008)] available today such situation is quite common. A typical example is the CD (compact disc) to DAT (digital audio tape) conversion, when a signal sampled with the sample rate $F_{s,CD} = 44.1$ kSa/s needs to be converted into a signal with the sample rate $F_{s,DAT} = 48$ kSa/s [Rajamani *et al.* (2000)].

The classic approach to the sample rate conversion (SRC) is presented in Fig. 1 [Mitra and Kaiser (1993)]. First, the input sample rate F_{s1} is increased L-times by inserting $L-1$ zeros between each two input samples. Next, a lowpass filter is used to remove spectral images located at multiples of the input sample rate. This replaces zeros, which have been previously inserted, with values of the interpolated input signal. Finally, a sample rate is decreased M-times by leaving only every M-th sample, thus signal with the sample rate $F_{s2} = L/MF_{s1}$ is obtained. Interpolation and decimation factors used in this process can be computed using the following

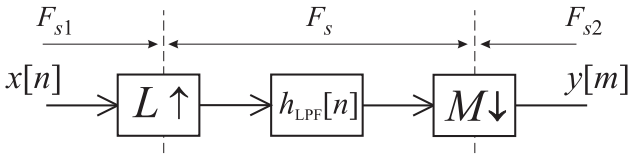

Fig. 1. Classic sample rate conversion algorithm.

formulas

$$L = F_{s2}/\gcd(F_{s1}, F_{s2}), \qquad M = F_{s1}/\gcd(F_{s1}, F_{s2}). \tag{1}$$

Such an approach to the SRC is relatively simple in implementation and interpretation but at the same time computationally inefficient. Nevertheless, computational efficiency can be readily improved with polyphase structures [Harris (2004)]. The more serious problem is that when L or M are about a hundred or larger, like in the aforementioned CD to DAT conversion with $L = 160$ and $M = 147$, design of the interpolation filter is problematic. The required transition band becomes very narrow and a very long impulse response of the interpolation filter is required. Therefore the main challenge in the SRC algorithm implementation is the interpolation filter design. A designer tries to obtain a shortest possible filter, which means lower computational costs and filter delay, fulfilling given specification described by passband ripples, stopband attenuation, and width and location of transition band. For very long impulse responses optimal solutions might not be reachable and less efficient filter design methods need to be used, like the window method. Moreover, when a ratio of input and output sample rates is an irrational number or when it varies in time, the factors L and M cannot be determined and the interpolation

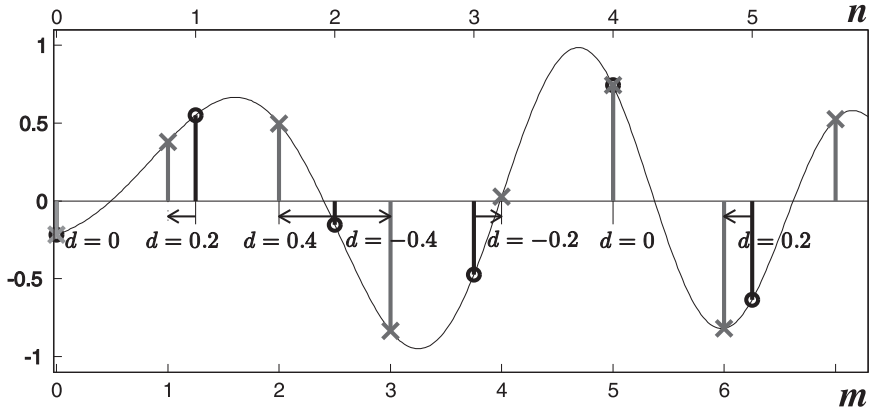

Fig. 2. Presentation of resampling process in the time domain. Black arrows show the delay between output samples and closest input samples. $L = 5$, $M = 4$.

filter cannot be specified.

Therefore, in practice, the classic sample rate conversion algorithm can be used only to resample a signal by factors which can be represented as a ratio of two relatively small integer numbers. For other resampling ratios a different approach to the resampling needs to be used.

Let us notice that the SRC algorithm actually has to compute values of signal samples in new time instants located between original samples (Fig. 2) [Rajamani *et al.* (2000); Hermanowicz *et al.* (2000); Evangelista (2003); Tarczynski *et al.* (1994)]. This means that we can treat each output sample of the SRC algorithm as an input sample delayed or advanced by a fraction of the input sampling period.

The fractional delay (FD) between the current output sample $y[m]$ and the nearest input sample $x[n]$ can be computed using the following recursive formula [Blok (2002a); Blok (2012b)]

$$d[m] = d[m-1] - F_{s1}/F_{s2} + \Delta n[m] \in [-0.5, 0.5) \tag{2}$$

where the resampling ratio $F_{s1}/F_{s2} = M/L$ and $\Delta n[m]$ is a number of new samples required in the input buffer for computation of the next output sample

$$\Delta n[m] = \text{round}(F_{s1}/F_{s2} - d[m-1]). \tag{3}$$

Using those two parameters we can formulate the resampling algorithm (Fig. 3):

(1) start with $d[0] = 0$ and $\Delta n[0] = 1$,
(2) wait for $\Delta n[m]$ new samples in the input buffer,
(3) compute the output sample $y[m]$ delayed by $d[m]$,

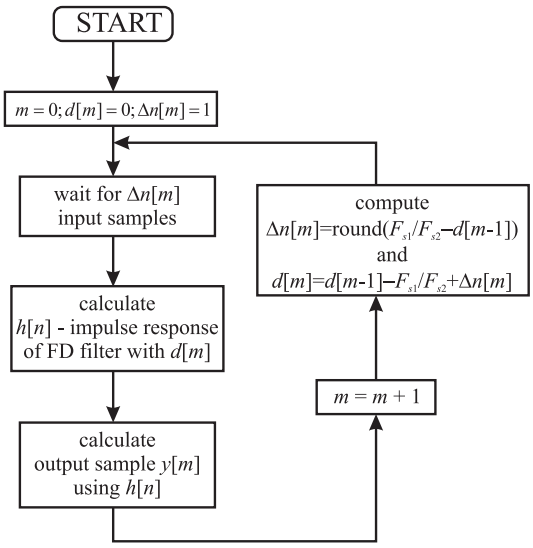

Fig. 3. Diagram of SRC algorithm based on VFD filter.

(4) calculate $\Delta n[m]$ and $d[m]$ for the next m and go back to step (2).

It is worth noticing that the resampling ratio F_{s1}/F_{s2} in (2) and (3) does not need to be rational and, moreover, it can change for each output discrete time instant m [Blok and Drózda (2012)]. To achieve this we need, however, to calculate a new set of filter coefficients for every output sample but computation of each output sample requires the FD filter with the impulse response approximately L-times shorter than the length of interpolation filter required in the classic approach.

2. Fractional delay filter

The ideal frequency response of the FD filter with the total delay τ_d is defined by the following formula [Laakso *et al.* (1996)]

$$H_{\mathrm{id}}(f) = \exp(-j2\pi f\tau_d), \quad f \in [-0.5, 0.5) \tag{4}$$

which corresponds to the ideal impulse response

$$h_{\mathrm{id}}[n] = \mathrm{sinc}(n - \tau_d). \tag{5}$$

Since the ideal impulse response is infinite and non-causal, in practical applications, the frequency response (4) must be approximated with an finite order filter. In this paper we will consider the approximation with the use of FIR FD filter with the frequency response

$$H_N(f) = \sum_{n=0}^{N-1} h[n] \exp(-j2\pi f n) \tag{6}$$

where $h[n]$ is the impulse response of the length N.

Because of the causality requirement, FD filters are usually characterized with a nonzero integer delay $D = \mathrm{round}(\tau_d)$, which for FIR filters is commonly selected close to the bulk delay $\tau_N = (N-1)/2$. With those two delays defined, we receive the following formula for the total delay

$$\tau_d = D + d = \tau_N + \varepsilon \tag{7}$$

where $d \in [-0.5, 0.5)$ is the fractional delay and ε is the net delay. In practice also the net delay ε is limited to $[-0.5, 0.5)$. With this assumption fractional delay d is equal to the net delay ε for odd N, while for even N the fractional delay $d = \varepsilon - \mathrm{sign}(\varepsilon)/2$ and has discontinuity at $\tau_d = \tau_N$ [Blok (2002b)].

The performance of the FD filter is usually evaluated using the frequency domain error function [Laakso *et al.* (1996)]

$$E(f) = H_N(f) - H_{\mathrm{id}}(f), \tag{8}$$

but it is not sufficient to know just the errors of FD filters to assess the performance of the SRC algorithm based on FD filters. Relations between all FD filters used in the resampling are also important. These relations can be readily taken into account if we observe that the SRC algorithm based on FD filters (Fig. 1.3) is

equivalent to the classic approach (Fig. 1) [Blok (2002a)]. We only need to replace the interpolation filter with the overall filter, obtained by polyphase composition of FD filters used in resampling. This can be done only for rational resampling rates but the conclusions resulting from the overall filter can be readily adapted to arbitrary resampling ratios.

The composition of the impulse response of the overall filter is defined by the following formula [Blok (2002a)]

$$h_o[m + nL] = h_{d[m]}[n]; \quad m = 0, 1, \ldots, L - 1 \tag{9}$$

where $h_{d[m]}[n]$ is the impulse response of the FD filter with fractional delay $d[m]$. In order to obtain a proper overall filter, delays $d[m]$ need to be organized in the decreasing order

$$d[m - 1] = d[m] + 1/L; \quad m = 1, \ldots, L - 1. \tag{10}$$

Using the overall filter we can readily analyze distortions introduced by SRC algorithm based on FD filters since this filter should fulfill the same design requirements as the interpolation filter in the classic approach (Fig. 1).

It is worth noting that SRC algorithms with different decimation factors M, but with the same interpolation factor L, operate on the same set of fractional delays. Therefore, since the same set of FD filters is used, the overall filter also stays the same. Nonetheless, we must remember that when the output sample rate is smaller than the input sample rate ($M > L$) the cutoff frequency f_c of the interpolation filter should be lower

$$f_c = \min(0.5/L, 0.5/M) \tag{11}$$

which must be taken into account in the SRC algorithm design.

3. FD filter design for SRC

Since the problem of fractional delay implementation occurs in many applications many FD filter design method are available [Laakso *et al.* (1996)]. The SRC based on FD filters has many advantages over the classic approach but its performance depends on a method used to design FD filters. The problem is that most of approaches to FD filter design are focused on performance of FD filter itself and do not concentrate on relation between FD filter design method and SRC algorithm performance. If the filter has to be used for the SRC such an approach is not always satisfactory. This paper is focused on presentation of how different approaches to FD filter design influence properties of the SRC algorithm based on FD filters. Based on observed properties of SRC algorithms we propose to organize FD filter design methods into three general categories: the optimal FD filter design, the offset window method and the polyphase decomposition. As we will present further in this paper, SRC algorithms based on FD filters belonging to each of these categories demonstrate different properties. In this paper only FIR FD filters are considered

since the design and analysis of the SRC algorithm based on such filters is simpler than in case of IIR filters.

3.1. *Optimal FD filters*

In the optimal FD filter design an error dependent on given criteria based on complex approximation error (8) is minimized. The most commonly used criteria are maximal flatness of error frequency response (MF) and minimization of least square error (LS)

$$E_{\mathrm{LS}}(f_a) = 2\int_0^{f_a} \left|E(f)^2\right| \mathrm{d}\, f \tag{12}$$

or maximum magnitude of approximation error (minimax)

$$E_{\mathrm{PE}}(f_a) = \max_{f\in[0,f_a]} |E(f)| \tag{13}$$

in a given approximation band defined by its upper frequency f_a [Laakso *et al.* (1996)]. The MF condition means that the error function has to be zero together with its $N-1$ consecutive derivatives.

The design of optimal FD filters is quite complex since it involves solving matrix equation [Laakso *et al.* (1996)]

$$\mathbf{Ph} = \mathbf{p} \qquad \Rightarrow \qquad \mathbf{h} = \mathbf{P}^{-1}\mathbf{p} \tag{14}$$

where $\mathbf{h}$ is a column vector of impulse response samples. The matrix $\mathbf{P}$ and vector $\mathbf{p}$ depend on the optimization criteria selected in the design. For example, in case of MF filter

$$P_{k+1,n+1} = n^k \quad \text{and} \quad p_{1,k+1} = \tau_d^k \tag{15}$$

while for LS filter

$$P_{k+1,n+1} = f_{\mathrm{a}}\,\mathrm{sinc}(f_{\mathrm{a}}(k-n)) \quad \text{and} \quad p_{1,k+1} = f_{\mathrm{a}}\,\mathrm{sinc}(f_{\mathrm{a}}(k-\tau_d)). \tag{16}$$

In both cases $n, k = 0, 1, \ldots, N-1$. Similarly, for minimax filter

$$P_{k+1,n+1} = \cos(2\pi f_{\mathrm{k}} n) - \sin(2\pi f_{\mathrm{k}} n), \quad P_{k,N+1} = (-1)^k \tag{17a}$$

and

$$p_{1,k+1} = \cos(2\pi f_{\mathrm{k}}\tau_d) - \sin(2\pi f_{\mathrm{k}}\tau_d). \tag{17b}$$

Here $n = 0, 1, \ldots, N-1$, $k = 0, 1, \ldots, N$ and the vector $\mathbf{h}$ has one additional element with magnitude equal to peak approximation error (13). Additionally, in this case of recursive algorithm has to be used to determine a set of extremal frequencies f_k [Laakso *et al.* (1996); Blok (2002b)], which means that the matrix equation (14) needs to be solved several times [Laakso *et al.* (1996); Blok (2002b)] which increases computational complexity.

The problem of high computational complexity of optimal FD filter design can be circumvented using the extracted window concept [Hermanowicz (1998); Blok

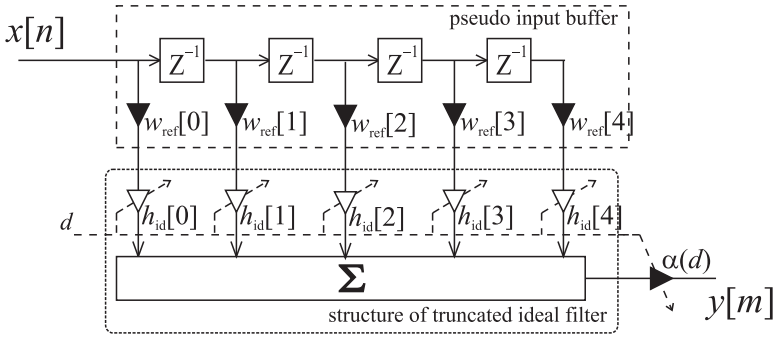

Fig. 4. Optimal FD filter structure based on the extracted window method

(2011a); Blok (2011b)]. In this approach the optimal VFD (variable fractional delay) filter is implemented using the window method design formula

$$h_d[n] = w_d[n]h_{\mathrm{id}}[n] \tag{18}$$

where window

$$w_d[n] = \alpha(d)w_{\mathrm{ref}}[n] \tag{19}$$

depends only on a single symmetric reference window $w_{\mathrm{ref}}[n]$ and a gain correction factor $\alpha(d)$. For optimal filter implementation using structure presented in Fig. 4 the reference window has to be equal to the even part of the window extracted, using reversed window method formula

$$w_{\mathrm{ext}}[n] = h_{\mathrm{opt}}[n]/h_{\mathrm{id}}[n], \tag{20}$$

from impulse response of the optimal filter $h_{\mathrm{opt}}[n]$ designed for an arbitrary selected fractional delay $d_{\mathrm{ref}} \neq 0$. The gain correction factor can be computed using the following formula

$$\alpha(d) = \frac{1}{\sum\limits_{n=0}^{N-1} \mathrm{sinc}(2f_a(n-\tau_d))w_{\mathrm{ref}}[n]h_{\mathrm{id}}[n]} \tag{21}$$

which is relatively complex but it can be readily approximated using low order polynomial [Blok (2011a)].

The additional advantage of VFD filter structure from Fig. 4 is that parameters of the resampling algorithm implemented using this structure can be readily modified during runtime. One only needs to replace a symmetric window $w_{ref}[n]$ and coefficients of polynomial approximating gain correction factor $\alpha(d)$. This makes such an approach well suited for SRC algorithms prototyping, when we need to verify which filter type or approximation band width should be selected in the final implementation or when a versatile application, which leaves the decision on selection of the filter type and its specification to the user, is needed.

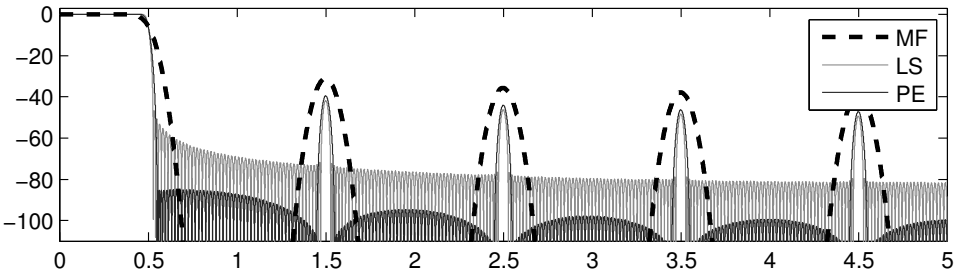

Fig. 5. Magnitude responses of overall filters composed of optimal FD filters. For FD filters of length $N = 51$: maximally flat, minimax with $f_a = 0.475$ and LS with $f_a = 0.45$. Interpolation factor $L = 10$. Frequency normalized by the input sample rate.

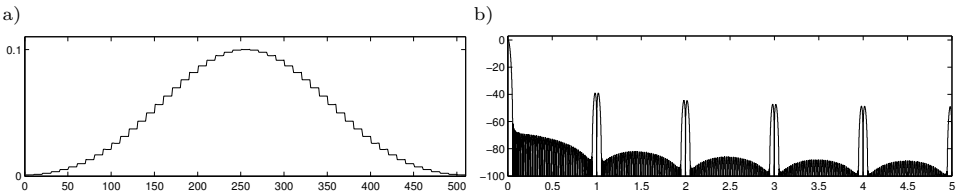

Fig. 6. Overall window (a) and magnitude response of overall window (b) composed of windows extracted from minimax FD filters of length $N = 51$ with $f_a = 0.475$. Interpolation factor $L = 10$.

Optimal FD filters might seem to be the best choice for the SRC since they offer the best possible approximation of the ideal FD filter for particular filter length and approximation band. Nevertheless, each filter is optimized separately, which means that relations between all the filters used in the resampling process are neglected. In the result magnitude response of the overall filter obtained from optimal FD filters exhibits large lobes in the stopband located at frequencies corresponding to components of the input signal located above f_a and at frequencies of images of those components (Fig. 5) [Blok (2002a)].

To source of this anomalies can be found if we investigate the overall window (Fig. 6a) composed of extracted windows (20) in the similar way as the overall filter is created. As we can see the overall window is characterized by the discontinuities which result in large lobes in stopband of the magnitude response of the overall window (Fig. 6b) which in turn result in large lobes in magnitude response of the overall interpolation filter (Fig. 5).

Thus, the SRC algorithm implemented using optimal filters performs correctly only for signals with band limited to the approximation band of FD filters used in the resampling. In Fig. 7 we can see the input signal component with linearly increasing frequency and distortions resulting from aliased spectral replicas of the resampled signal. These distortions are spread over the whole

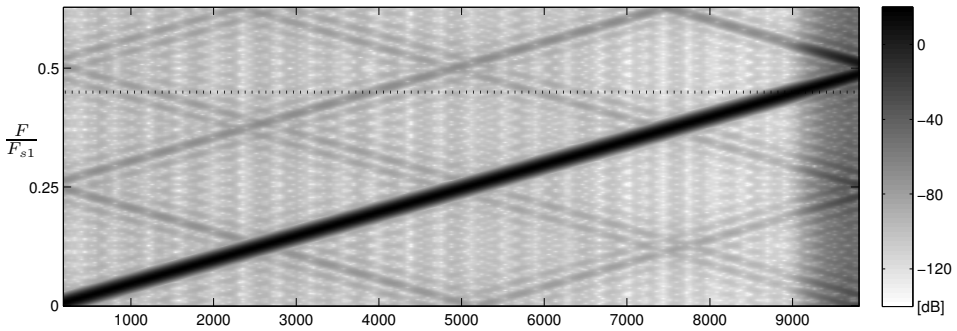

Fig. 7. Spectrogram of fullband chirp signal with sample rate increased by $2\pi/5$ using minimax FD filters of length $N = 51$ with $f_a = 0.475$. Dotted line marks f_a frequency.

frequency range but are below -80dB as long as the normalized frequency of the input chirp signal is smaller than f_a. When the input signal frequency exceeds f_a the distortions level significantly increases. Therefore, for fullband signals the additional filter preceding the actual resampling, limiting the band of the input signal, is required. Such a prefilter increases computational cost of the resampling but with multistage implementation [Rajamani *et al.* (2000); Hermanowicz (2004)] which improves computational efficiency of the resampling, the prefiltering can be readily incorporated into the interpolation filter of the first resampling stage.

3.2. *FD filter design using offset window*

The second approach to the FD filter design, similarly to the extracted window method used in implementation of optimal FD filters, is based on the window method (18). The difference is that the symmetric reference window is offset accordingly to the fractional delay of the designed filter [Cain *et al.* (1995); Cain *et al.* (1996); Yardim *et al.* (1997); Blok (2012a)]

$$w_d[n] = w((n - d)T_{s1}). \tag{22}$$

Thus the offset window is a sequence of samples of symmetric continuous prototype time window $w(t)$ sampled with sampling period $T_{s1} = 1/F_{s1}$ in instants delayed by d.

The important property of the FD filter design using the window offsetting method is the fact that the overall window is simply the L times interpolated window of the same type as the prototype window used to design FD filters. Thus, using FD filters designed with offset window we actually design overall filter using window method while designing only a fraction of the whole filter with each FD filter. Additionally, since the overall window is smooth (Fig. 8a), the magnitude

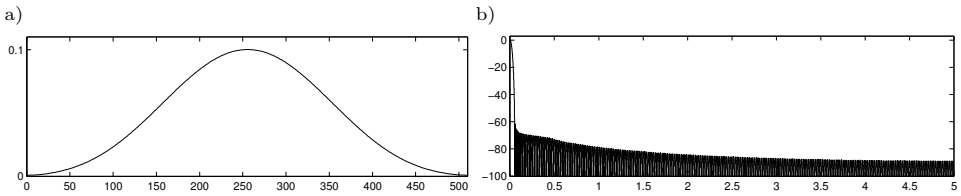

Fig. 8. Overall window (a) and magnitude response of overall window (b) composed of offset windows with reference window extracted from minimax FD filters of length $N = 51$ with $f_a = 0.475$. Interpolation factor $L = 10$.

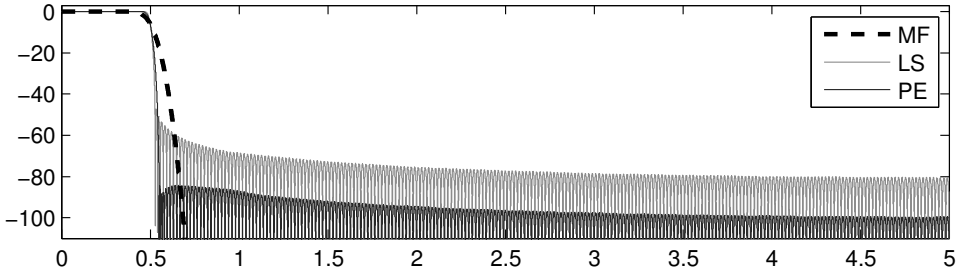

Fig. 9. Magnitude responses of overall filters for FD filters designed using offset window method with window extracted from minimax (PE) ($f_a = 0.45$), LS ($f_a = 0.475$) and MF FD filters. $N = 51$ and $L = 10$.

response of the overall window does not exhibit large lobes in stopband (Fig. 8b) which are typical to the overall window composed for the SRC algorithm based on optimal FD filters (Fig. 6). In order to achieve best resampling results the scale of the prototype window should be adjusted in such a way that the average gain of the overall filter in the passband is equal to L.

The offset window method is not optimal and separate FD filters designed using offset windows demonstrate performance worse than the performance of optimal filters. Nevertheless, the overall interpolation filter performs better in the stopband since it does not exhibit large lobes there (Fig. 9) [Blok (2012a)]. This means that the SRC algorithm based on FD filters designed with offset window method can be used in the resampling of fullband signals without need for a prefilter. For example if we compare Fig. 10 with Fig. 7 we can see that when chirp frequency is below f_a the distortions for offset window method and optimal filters are similar. The significant difference can be observed when chirp frequency is greater than f_a. For offset window method only residues of the first spectral replica remain with other replicas suppressed.

Additionally, using this approach we can readily manipulate the location of the transition band of the interpolation filter. Let us notice that the impulse response

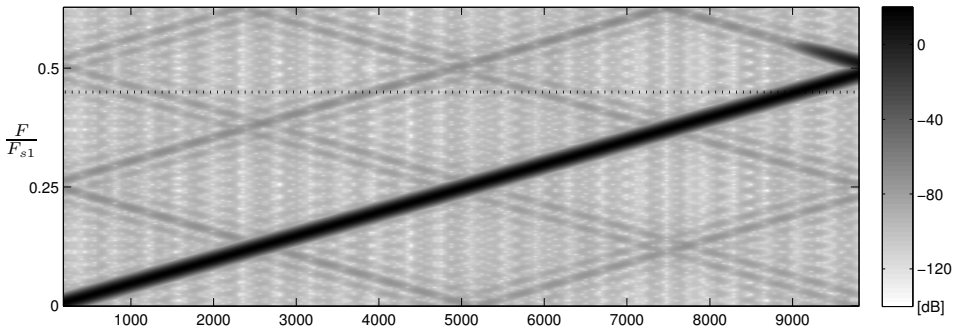

Fig. 10. Spectrogram of fullband chirp signal with sample rate increased by $2\pi/5$ using FD filters designed using offset window with reference window extracted from minimax FD filters of length $N = 51$ with $f_a = 0.45$.

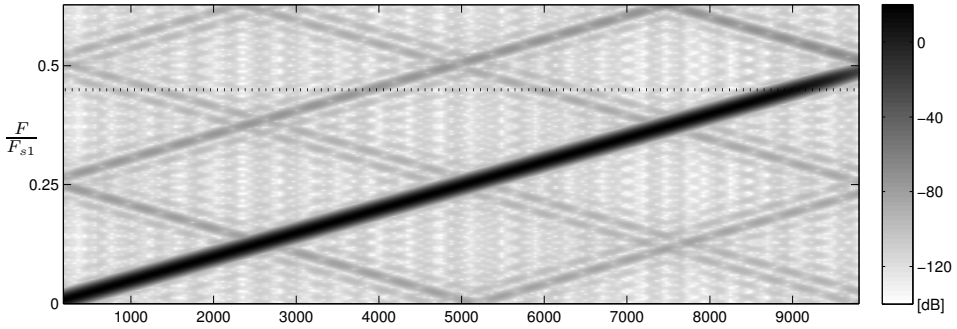

Fig. 11. Spectrogram of fullband chirp signal with sample rate increased by $2\pi/5$ using lowpass FD filters designed using offset window with reference window extracted from minimax FD filters of length $N = 51$ with $f_a = 0.45$.

of the overall filter composed of truncated impulse responses of the ideal FD filter (5) is the truncated ideal response of the $1/L$-th band interpolation filter

$$h_{\mathrm{LPF,id}}[n] = \mathrm{sinc}(2f_c n) \tag{23}$$

where cutoff frequency $f_c = 0.5/L$ and the gain of the filter is equal to L. To change cutoff frequency of overall interpolation filter (23) the impulse response of the fullband FD filter (5) needs only be replaced with ideal impulse response of the band-limited FD filter

$$h_{\mathrm{id}}[n] = \mathrm{sinc}(Lf_c(n - \tau_d)). \tag{24}$$

This helps to limit distortions inherent to SRC, for example, in Fig. 11 the attenuation of spectral replicas of high frequency components has been significantly improved in comparison to the use of fullband FD filters (Fig. 10).

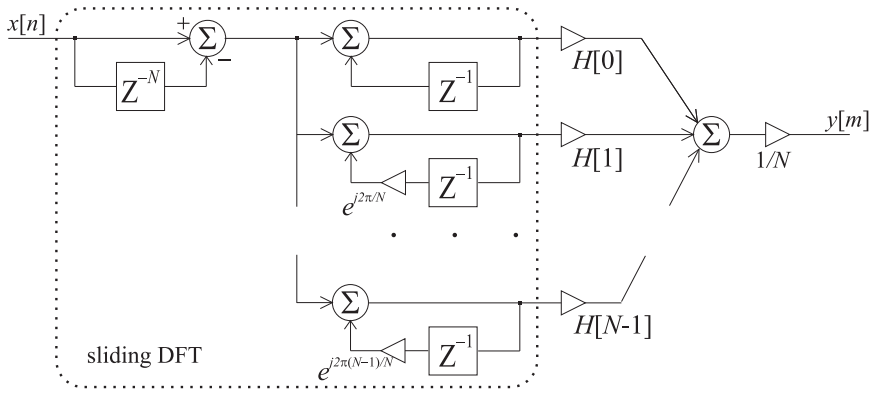

Fig. 12. Frequency sampling FIR filter structure for implementation of VFD filter with impulse response N based on its N-point frequency response $H[k]$.

The concept of the design based on offset window is simple when a prototype window $w(t)$ is given as a continuous function of time, like in case of raised cosine windows [Albrecht (2001)]. The problem with raised cosine windows is, that even with optimized coefficients, filter designed using such a window is worse than the optimal filter by few dB. On the other hand, offsetting other types of windows is more problematic. For example, the Kaiser window [Mitra and Kaiser (1993)] is defined as a continuous function of time variable but coefficients recalculation for different delays is numerically too demanding for most real time applications of the SRC. If the window is not defined in the time domain as a continuous function, like the Chebyshev window [Mitra and Kaiser (1993)] which is defined in the frequency domain, then in addition to coefficients calculation formula being too complex, the window offsetting procedure is not straightforward.

Offsetting procedures for such windows are based on the fact that window offsetting can be interpreted as delaying a discrete symmetric prototype window by a fraction of the sampling period. Thus, when a prototype window is not a continuous function, it might be resampled using a delay operator implemented either in the frequency domain [Laakso *et al.* (1995); Pei and Lai (2012)] or in the time domain, e.g. using short FD filter [Blok (2012a)].

If we want to decrease computational costs of window offsetting even more, after computing several offset windows for different delays, the discrete window prototype can be approximated with piecewise polynomial. Precise window offsetting can be then performed in time domain with separate polynomials approximating window segments corresponding to each sample of impulse response of the designed FD. Usually second or third order polynomials offer a sufficient performance [Blok (2012c)].

Another solution is to implement FD filter in DFT domain using frequency sampling structure of FIR filter. Such implementation utilizing sliding DFT (Fig. 12)

and CORDIC hardware for computation of FD filter frequency response is even more efficient VFD filter structure than commonly used Farrow structure [Selva (2008); Blok (2013a)]. The additional advantage is that this structure is directly related to offset window method allowing for computationally efficient implementation of offset FD filters with nearly optimal performance [Blok (2013a)]. Moreover, with this structure, apart from the fractional delay, the filter bandwidth can also be adjusted without need for new set of coefficients [Blok (2013b)]. This property is especially important in SRC with variable resampling ratio which requires a decrease of the overall filter bandwidth when sample rate is decreased [Blok and Drózda (2013)]. However, we cannot use the extracted window with this structure and has to find coefficients of a raised cosine window [Blok (2013a); Blok (2013b)].

3.3. *Polyphase subfilters*

The last category of FD filter design methods suited for the needs of the SRC is directly related to the classic approach. Let us notice that a polyphase decomposition of $1/L$-th band optimal interpolation filter [Harris (2004)], designed for example using the Parks-McClellan algorithm, into L subfilters

$$h_{d[m]}[n] = h_{LPF}[m + nL]; \quad m = 0, 1, \ldots, L - 1 \tag{25}$$

gives us L fullband FD filters, each with different delay $d[m]$. With those filters stored in memory we can implement the SRC algorithm with interpolation factor L. The problem is that for large L we need to design and store very long impulse response of the interpolation filter, in some cases even longer than several thousands samples. Such optimal filter design might not be possible due to accumulation of numerical errors during design. Additionally, a particular interpolation filter can only be used for the resampling with the given factor L. On the other hand, for a given length of the overall filter we gain a possibility to improve the attenuation in stopband of the overall filter in the exchange for increased ripples in passband

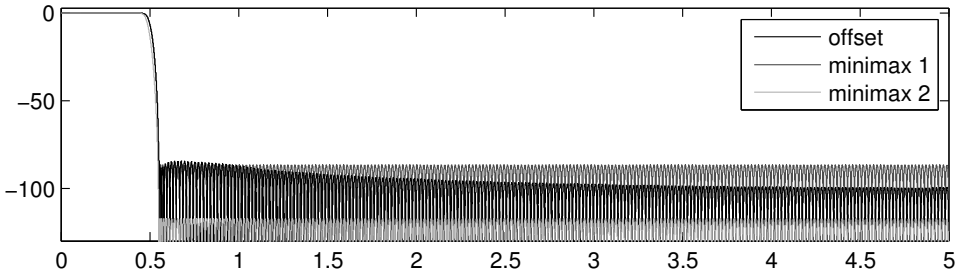

Fig. 13. Magnitude responses of overall filters combined from FD filters designed using offsetting of window extracted for minimax FD filter (from Fig. 9) (offset) and minimax interpolation filters with the same transition band and impulse response length designed with similar passband ripple level like previous one (minimax 1) and with larger passband ripple level (minimax 2).

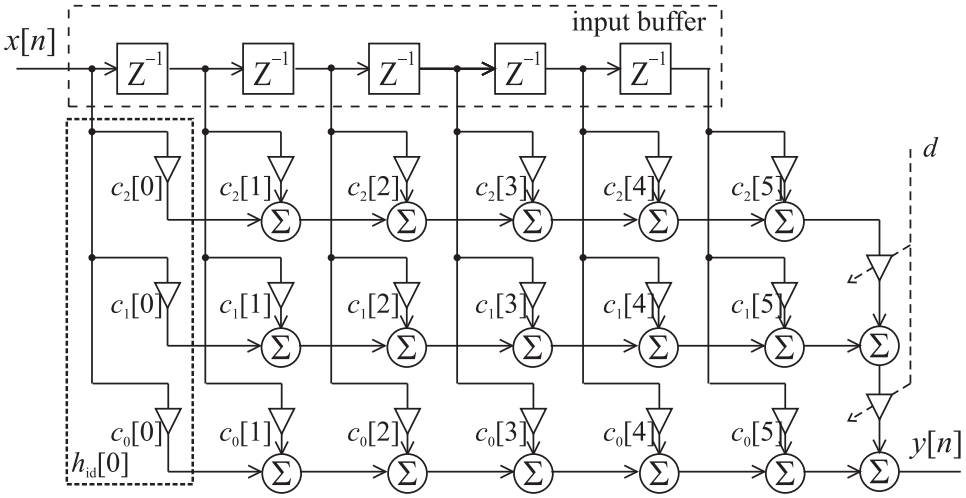

Fig. 14. Farrow structure of order $p = 2$ approximating the FD filter of the length $N = 6$. Thick dashed box indicates structure coefficients $c_m[0]$ of the polynomial approximating the first sample of the impulse response $h_{\mathrm{id}}[0]$.

(Fig. 13). This is a significant advantage. For example with stopband attenuation equal to 86 dB (Fig. 13) with two previous approaches to the FD filter design, when no prefilter is used, passband ripples of the overall filter are about 10^{-3} dB. In most application we don't need such a high precision in passband. Relaxing the specification in passband and allowing ripples equal to 0.02 dB, improves stopband attenuation by 30 dB (Fig. 13) for the same filter length.

This approach might seem inappropriate for incommensurate or variable resampling ratios, since we get only L FD filters, but we actually don't need to directly design the interpolation filter for the required ratio. We only need to design the prototype interpolation filter for some low integer interpolation factor, for example $L = 10$ (Fig. 13). Subsequently, using the Farrow structure [Hermanowicz (2004); Farrow (1988); Blok (2005); Harris (1997)] (Fig. 14) we can approximate the FD filter and obtain the impulse response for any required fractional delay.

The idea behind the Farrow structure is that the overall filter impulse response is approximated with a low order piecewise polynomial with each segment approximating separate sample of the FD filter impulse response

$$h[n] = \sum_{m=0}^{p} c_m[n] d^m. \tag{26}$$

Thus using the Farrow structure we can keep the advantages of the SRC algorithm specification flexibility resulting from of the direct design of the interpolation filter and at the same time implement any arbitrary resampling ratio (Fig. 15). That way we can freely select cutoff frequency and adjust passband ripple level

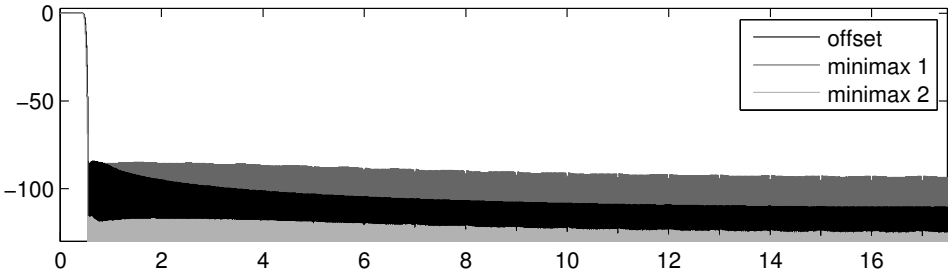

Fig. 15. Magnitude responses of overall filters obtained using the Farrow structure for interpolation factor $L = 35$. Farrow structure coefficients computed based on filters presented in Fig. 13. The length of the impulse response of FD filters $N = 51$.

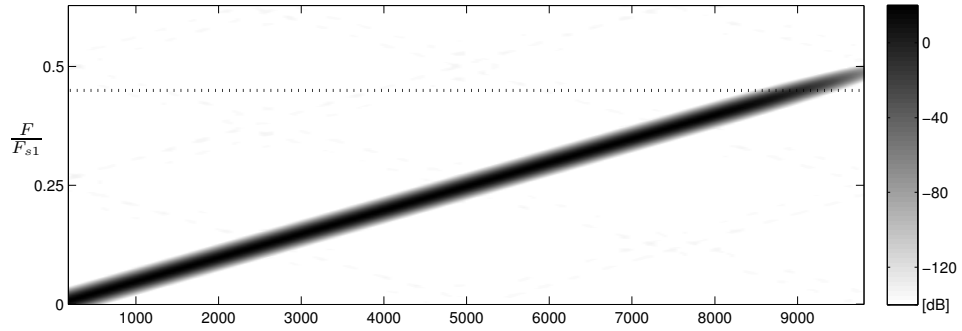

Fig. 16. Spectrogram of fullband chirp signal with sample rate increased by $2\pi/5$ using FD filters obtained by polynomial approximation of polyphase filters of length $N = 51$.

while improving the stopband attenuation. However, for each different specification we need to replace all coefficients of the Farrow structure. It is worth noting that although the Farrow structure can be used to implement the VFD filter belonging to any category described in this paper, the structure is the most beneficial when used with polyphase filters obtained from the lowpass prototype of the interpolation filter.

An example of the resampled chirp signal for incommensurate resampling ratio is presented in Fig. 16. We can see that attenuation of all components of replicas replicas have been significantly improved without need for filter length increase. Nonetheless, this have been achieved at the cost of larger distortions in the band of the signal resulting from increased passband ripples of the prototype interpolation filter.

3.4. *Comparison*

The properties of SRC algorithm based on FD filters are different for each presented category of methods for FD filter design. For filters from the first category, optimal filters, the overall filter of the SRC algorithm exhibits large lobes in the stopband. This means that either the input signal must be band-limited or a prefilter needs to be used before resampling. Since FD filters belonging to this group are closely related to the symmetric window design method, such an adjustable FD filter can be implemented using extracted window method. This implementation allows for simple change window type or width of the approximation band which might be useful in some applications.

If the properties of the overall filter in the stopband have to be improved, the offset window method should be used to design FD filters instead of the optimal filters. This results in elimination of large lobes in stopband and additionally the cutoff frequency of the overall filter can be readily changed. A design of FD filters using the offset window method when compared with design of optimal filters with symmetric extracted window is more computationally complex but the SRC algorithm based on the offset window approach does not require additional prefilter since large lobes in stopband of the overall filter are suppressed.

The last category includes FD filters designed by means of polyphase decomposition of the interpolation filter. With this approach we can directly optimize the interpolation filter which means that a designer can balance between passband ripple and stopband attenuation. This allows for trade-off between in-band distortions and spectral replicas attenuation. In this approach using polyphase decomposition we obtain only few FD filters from the whole family but the Farrow structure can be used to obtain FD filters with any required fractional delay which makes an implementation of the arbitrary resampling ratio possible using FD filters from this group. Unlike with previous categories, polyphase filters and Farrow structure coefficients need to be redesigned each time we want to change specification of the SRC algorithm. This is the price we have to pay for the flexibility of the overall filter specification.

Time domain errors are similar for optimal FD filters and offset window based fullband FD filters (Fig. 17a i b). The difference is that for offset windows based filters the error increase at the end of the signal is the result only from change of amplitude of the resampled chirp, while for optimal filters apart for the amplitude change, aliasing with spectral replicas occurs (Fig. 7 and 10). In Fig. 17c time domain error start to increase earlier. This is the result of transition band of the overall filter shifted towards lower frequencies. And even though there are no additional frequency components in the resampled signal (Fig. 11), the amplitude change of the resampled chirp begins earlier, at lower frequency. In case of FD filters obtained using polyphase decomposition the spectral replicas attenuation have been increased (Fig. 16) but this results in increased passband ripples which means that distortions of the amplitude of the resampled chirp are larger. Therefore, as

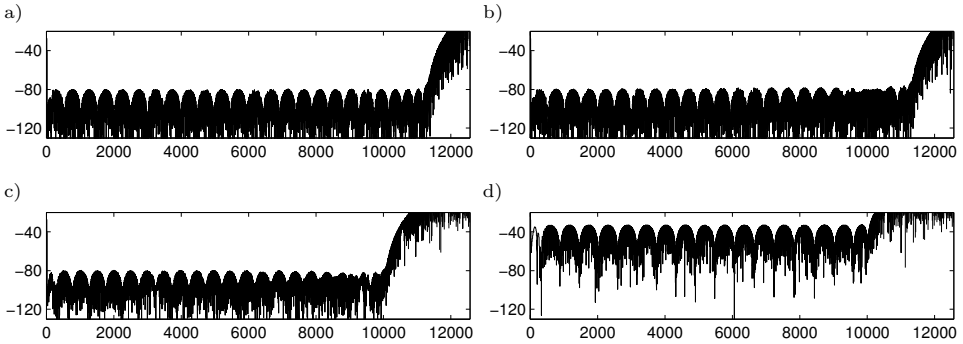

Fig. 17. Time domain resampling error (in dB) for signals from Fig. 7 (a), Fig. 10 (b), Fig. 11 (c) and Fig. 16 (d).

we can see in Fig. 17, the overall error in time domain is larger for this approach to FD filter design.

To decrease computational complexity of VFD filter implementation the Farrow structure (Fig. 12) can be used for all discussed here FD filter design methods. Such implementation requires $(p+1)N$ real additions and multiplications per output sample, where p is Farrow structure order. This can be halved if structure coefficients symmetry is exploited. For example for all filters presented in this paper with $N = 51$ we can use $p = 6$, which gives approximately 179 additions and multiplications per output sample. Alternatively, for DFT domain implementation of VFD filter which is equivalent to offset window method, a frequency sampling FIR filter structure can be used. For real valued input signal this structure approximately requires $(\frac{3F_{\mathrm{p1}}}{2F_{\mathrm{p2}}}+2)N$ real additions and $(\frac{2F_{\mathrm{p1}}}{F_{\mathrm{p2}}}+3)N$ real multiplications per output sample plus one sinus and one cosine calculation. For $\frac{F_{\mathrm{p2}}}{F_{\mathrm{p1}}} = 2\pi/5$ and $N = 51$ this gives 163 additions and 234 multiplications per output sample. Since DFT of the input signal computed is computed in this structure using sliding DFT is updated for each input signal, while frequency response of the FD filter and DFT of the output signal are computed for each output sample, the computational efficiency of this structure is better for larger resampling ratios $(\frac{F_{\mathrm{p2}}}{F_{\mathrm{p1}}})$. If the CORDIC hardware is available approximately only $(\frac{F_{\mathrm{p1}}}{2F_{\mathrm{p2}}}+\frac{1}{2})N$ real multiplications and $(\frac{F_{\mathrm{p1}}}{2F_{\mathrm{p2}}}+1)N$ CORDIC rotations are required. For $\frac{F_{\mathrm{p2}}}{F_{\mathrm{p1}}} = 2\pi/5$ and $N = 51$ this gives only 46 additions and 71 CORDIC rotations per output sample.

4. Conclusions

This paper presents an overview of existing FD filter design methods in perspective of properties of the SRC algorithm based on such filters. The dissimilarities of different methods of the FD filter design have been analyzed using the overall filter or the overall window. Based on the observed properties of SRC algorithms we have

proposed classification of FD design algorithms into three general categories: optimal design, which is related to symmetric window method, offset window method and polyphase decomposition. All these approaches are characterized with comparable computational complexity of VFD filter implementation but the most flexible approach for fixed resampling ratio and filter specification offers polyphase decomposition in which a designer can independently select transition band position and ripples in passband and stopband of the overall filter. The optimal filters offer the smallest possible approximation error but are suitable only for resampling of band-limited signals. The offset window method based FD filters, like polyphase filters allow for adjustment of transition band of the overall filter but the error ripples in passband and stoband cannot be selected independently. Nevertheless the advantages of offset window approach over the polyphase decomposition is that the change of the overall filter transition band location can be performed without need for filter redesign.

Acknowledgments

This work was in part supported by the Polish Ministry of Science and Higher Education under the research project financed from the state budget designated for science in the years 2010-2012.

References

AES5-2008 (2008). *AES Recommended Practice for Professional Digital Audio – Preferred Sampling Frequencies for Applications Employing Pulse-Code Modulation*, Audio Engineering Society, Inc.

Albrecht, H. (2001). A family of cosine-sum windows for high-resolution measurements. In Proc. ICASSP'01, **5**, pp. 3081–3084.

Blok, M. (2002). Collective filter evaluation of an FSD filter-based resampling algorithm. In OSEE 2002, http://www.eetimes.com/design/signal-processing-dsp/4017905.

Blok, M. (2002). Optimal fractional sample delay filter with variable delay. In OSEE 2002, http://www.eetimes.com/design/signal-processing-dsp/4018005.

Blok, M. (2005). Farrow structure implementation of fractional delay filter optimal in Chebyshev sense. In Proc. SPIE, **6159**, p. 61594K.

Blok, M. (2011). On practical aspects of optimal FSD filter design using extracted window method. In Proc. ECCTD'2011, pp. 330–333.

Blok, M. (2011). Versatile structure for variable fractional delay filter based on extracted window method. In Proc. PWT'2011, pp. 64–67.

Blok, M. (2012). Fractional delay filter design with extracted window offsetting. In Proc. MixDes'2012, pp. 489–494.

Blok, M. (2012). Fractional Delay Filter Design for Sample Rate Conversion. In Proc. FedCSIS'2012, pp. 701–706.

Blok, M. (2012), Sample rate conversion with fractional delay filters designed using windows offset by means of polynomial interpolation. Telecommun. Rev. Telecommun. News (Prz. Telekomun. Wiad. Telekomun.), **4**, in Polish.

Blok, M. (2013). Comments on ,,Closed form variable fractional time delay using FFT". IEEE Signal Process. Lett., **20**, pp. 747–750.

Blok, M. (2013). Fractional delay filter with adjustable bandwidth implemented in DFT domain. In Polish, unpublished.

Blok, M.; Drózda (2012). Sample rate conversion with fluctuating resampling ratio. In Proc. NTAV/SPA, pp. 209–214.

Blok, M.; Drózda (2013). Variable Ratio Sample Rate Conversion Based on Fractional Delay Filter. Unpublished.

Cain, G.D.; Yardim A.; Henry P. (1995). Offset windowing for FIR fractional-sample delay. In Proc. IEEE ICASSP'95, pp. 1276–1279.

Cain, G.D.; Yardim, A.; Henry, P. (1996). Optimal two-term offset windowing for fractional delay. Electron. Lett., **32**, pp. 526–527.

Evangelista, G. (2003). Design of digital systems for arbitrary sampling rate conversion. Signal Process., **83**, pp. 377–387.

Farrow, C.W. (1988). A continuously variable digital delay element. In Proc. IEEE ISCAS'88, pp. 2641–2645.

Harris, F.J. (1997). Performance and design of Farrow filter used for arbitrary resampling. In Proc. DSP'97, **2**, pp. 595–599.

Harris, F.J. (2004), *Multirate Signal Processing for Communication Systems*, Prentice Hall, Upper Saddle River.

Hermanowicz, E. (1998). A nearly optimal variable fractional delay filter with extracted Chebyshev window. In Proc. IEEE ICECS'98, **2**, pp. 401–404.

Hermanowicz, E. (2004). On designing a wideband fractional delay filter using the Farrow approach. In Proc. EUSIPCO'2004, pp. 961–964.

Hermanowicz, E.; Rojewski, M.; Blok, M. (2000). A sample rate converter based on a fractional delay filter bank. In Proc. ICSPAT 2000, pp. 16–19.

Laakso, T.I., *et al.* (1996). Splitting the unit delay — tools for fractional delay filter design. IEEE Signal Process. Mag., **13**, pp. 30–60.

Laakso, T.I.; Saramäki, T.; Cain, G.D. (1995). Asymmetric Dolph-Chebyshev, Saramäki, and transitional windows for fractional delay FIR filter design. In Proc. MWSCAS'95, pp. 580 – 583.

Mitra, S.K.; Kaiser, J.F. (1993). *Handbook for Digital Signal Processing*, John Wiley & Sons, New York.

Pei, S.-C.; Lai, Y. (2012). Closed form variable fractional time delay using FFT. IEEE Signal Process. Lett., **19**, pp. 299–302.

Rajamani, K.; Yhean-Sen, L.; Farrow, C.W. (2000). An efficient algorithm for sample rate conversion from CD to DAT. IEEE Signal Process. Lett., **7**, pp. 288–290.

Selva, J. (2008). An efficient structure for the design of variable fractional delay filters based on the windowing method. IEEE Trans. Signal Process., **56**, pp. 3770–3775.

Tarczynski, A.; Kozinski, W.; Cain, G.D. (1994). Sampling rate conversion using fractional-sample delay. In Proc. IEEE ICASSP'94, pp. 285–288.

Yardim, A.; Cain, G.D.; Lavergne, A. (1997). Performance of fractional-delay filters using optimal offset windows. In Proc. IEEE ICASSP'97, **3**, pp. 2233–2236.