Implementation of Linear-Phase FIR Filters for a Rational Sampling-Rate Conversion Utilizing the Coefficient Symmetry

Citation
Bregovic, R., Yu, Y. J., Saramäki, T., & Lim, Y. C. (2011). Implementation of Linear-Phase FIR Filters for a Rational Sampling-Rate Conversion Utilizing the Coefficient Symmetry. *IEEE Transactions on Circuits and Systems. Part 1: Regular Papers*, *58*(3), 548-561. https://doi.org/10.1109/TCSI.2010.2072350

Year
2011

Version
Peer reviewed version (post-print)

Link to publication
TUTCRIS Portal (http://www.tut.fi/tutcris)

Published in
IEEE Transactions on Circuits and Systems. Part 1: Regular Papers

DOI
10.1109/TCSI.2010.2072350

R. Bregović, Y. J. Yu, T. Saramäki, and Y. C. Lim, “Implementation of linear-phase FIR filters for a rational sampling-rate conversion utilizing the coefficient symmetry,” *IEEE Trans. Circuits Syst. I*, vol. 58, pp. 548–561, March. 2011.

Implementation of Linear-Phase FIR Filters for a Rational Sampling Rate Conversion Utilizing the Coefficient Symmetry

Robert Bregović, *Member, IEEE,* Ya Jun Yu, *Senior Member, IEEE*, Tapio Saramäki, *Fellow, IEEE,* and Yong Ching Lim, *Fellow, IEEE*

Abstract— **This paper proposes an efficient structure for implementing a linear-phase FIR filter of an arbitrary order *N* for the sampling rate conversion by a rational factor of *L*/*M*, where *L* (*M*) is the integer up-sampling (down-sampling) factor to be performed before (after) the actual filter. In this implementation, the coefficient symmetry of the linear-phase filter is exploited as much as possible and the number of delay elements is kept as low as possible while utilizing the following facts. When increasing (decreasing) the sampling rate by a factor of *L* (*M*), only every *L*th input sample has a non-zero value (only every *M*th output sample has to be evaluated). In this way, the number of required multiplications per output sample is reduced approximately by a factor of 2 compared to the conventional polyphase implementation. The proposed implementation is first illustrated using two examples. Based on these examples, guidelines are then given on how to efficiently realize an *N*th-order linear-phase FIR filter for a sampling rate converter having an arbitrary rational conversion factor *L*/*M*. Finally, the implementation complexity of the proposed approach is evaluated and some examples are included showing the efficiency of the proposed implementation compared to other existing ones.**

Index terms—**Multirate system, rational sampling rate conversion, FIR filter, linear-phase**

I. INTRODUCTION

Many modern signal processing algorithms require more than one sampling rate in order to increase the efficiency of the overall system [1]–[8]. Such systems utilizing multiple sampling rates are known as multirate systems. The basic building blocks of a multirate system are interpolators by an integer factor *L* and decimators by an integer factor *M*. Combining these two blocks results in a system that changes the sampling rate by a rational factor *L*/*M*. The block diagram for such a rational sampling rate converter is depicted in Figure 1 [1]. As shown in this figure, sampling rate conversion by a rational factor *L*/*M* means that the signal is first up-sampled by a factor of *L*, the resulting up-sampled sequence is filtered by a transfer function $H(z)$, and, finally, the filtered signal is down-sampled by a factor of *M*. The filter with the transfer function $H(z)$, as shown in Figure 1, has the following two purposes: First, it suppresses the image spectra that are the results of up-sampling the input sequence by an integer factor *L*, and, second, it prevents the aliasing effects that occur after down-sampling by an integer factor *M*.

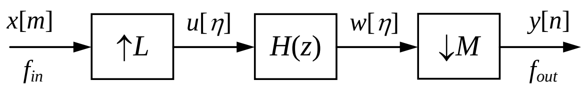


Figure 1. Rational sampling rate converter by a factor of *L*/*M*.

For the system of Figure 1, the relation between the output sampling rate, denoted by f_{out}, and the input sampling rate, denoted by f_{in}, is given by

$$f_{out} = (L/M) f_{in} . \qquad (1)$$

Depending on whether *M* or *L* is larger, the sampling rate converter of Figure 1 primary acts as a decimator or interpolator, respectively.

This paper concentrates on deriving an efficient implementation of the rational sampling rate converter shown in Figure 1 only when the *N*th-order transfer function $H(z)$ is given by

$$H(z) = \sum_{k=0}^{N} h_k z^{-k} , \qquad (2a)$$

where

$$h_{N-k} = h_k \quad \text{for} \quad k = 0,1,\ldots,N . \qquad (2b)$$

The $H(z)$ selected in the above manner is a transfer function of a linear-phase finite-impulse response (FIR) filter that possesses a symmetric impulse response, that is, only[1] $\lfloor N/2 \rfloor + 1$ of the filter coefficients are distinct.

The motivation for concentrating on this type of rational sampling rate converters lies in the fact that there are many applications utilizing these converters, e.g., software radio,

Manuscript received March 22, 2010; revised May 31, 2010. This work was supported by the Academy of Finland, project No. 213462 (Finnish centre of Excellence program (2006–2011)) and Temasek Laboratories @ Nanyang Technological University. Some parts of the material of this paper were presented at the IEEE International Symposium on Circuits and Systems, Island of Kos, Greece, May 2006 [18].

R. Bregović and T. Saramäki are with the Department of Signal Processing, Tampere University of Technology, P.O. Box 553, 33101 Tampere, Finland. e-mail: bregovic@cs.tut.fi and ts@cs.tut.fi.

Ya Jun Yu and Yong Ching Lim are with the School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798. e-mail: eleyuyj@pmail.ntu.edu.sg and elelimyc@pmail.ntu.edu.sg.

[1] $\lfloor x \rfloor$ stands for the largest integer that is smaller than or equal to *x*.

image resizing, digital audio re-sampling, medical applications. In most cases linear-phase FIR filters are used. This is because they do not possess feedback loops and, therefore, they are easier to implement in a multirate system when compared to IIR filters. Moreover, these converters have exactly linear phase in the frequency band of interest [9]. Due to this linear-phase property, the input-output relation of such a rational sampling rate converter does not suffer from any phase distortion. This is very important in many applications, where the envelopes of the time waveforms of the signals whose sampling rates are changed by a rational factor are desired to be preserved. A typical example is the processing of an electrocardiogram signal, for which the preservation of the original waveform, after the sampling rate conversion, is of a crucial interest for still being capable of performing diagnostics concerning heart diseases. Since most of the above-mentioned applications are used in real time, it is important to have an efficient implementation of all parts of the system, including the rational sampling rate converter.

For a system shown in Figure 1 that changes the sampling rate by a rational factor L/M, the implementations proposed in [1]–[4], [10]–[15] utilize the fact that the filter is between an up-sampler (only every Lth sample is non-zero) and a down-sampler (only every Mth sample is taken as the output). In these approaches, various techniques have been applied in order to reduce the implementation complexity of the overall system. For example, in [10]–[12] a polyphase structure is derived for efficiently implementing the filter (in the rest of the paper this approach is referred to as the polyphase implementation), whereas in [13] a fast Fourier transform based cyclic algorithm has been used. Furthermore, as various structures introduce different delays to the signal, in [14], [15] the emphasis has been put on structures that minimize the overall delay between the input and the output of the rational sampling rate converter. However, even if in most cases a linear-phase filter is used for implementing the rational sampling rate converters under consideration, in all of the above mentioned approaches, the coefficient symmetry of this filter has not been exploited in the implementation. Consequently, neither of those methods could achieve a reduction in number of multiplications per output sample by a factor of $2L$ compared to the direct implementation. Achieving such a reduction is the goal of this paper.

For either up-sampling or down-sampling with integer sampling rate conversion factors, efficient implementation structures exploiting the coefficient symmetries of a linear-phase FIR filter have been proposed in [16], [17]. When implementing these filters, in the case of interpolators and decimators, in order to minimize the number of required multiplications per output (input) sample, the coefficient symmetry of the filters as well as the fact that the filters are preceded by an up-sampler by L or followed by a down-sampler by M is taken into consideration.

The design approach proposed in [17] has been adapted in [18] for implementing rational sampling rate converters for filter orders $N = M(L-1)+kL$ with k being an integer. By using the method proposed in [18] the number of multipliers required in the implementation has been reduced compared with other existing approaches. In this paper, the ideas presented in [18] are extended further in order to enable an efficient implementation of rational sampling rate converters with linear-phase FIR filters for any combination of L, M, and N. It should be pointed out that it has been hinted in [17] that an approach similar to the one discussed there for interpolators and decimators can also be applied for rational sampling rate converters. However, as it is shown in this contribution, this is not as straightforward as expected due to the interaction between parameters L, M, and N.

In this paper, the efficiency of the proposed implementation method is compared with the methods discussed above. Beside these methods, that directly implement the system shown in Figure 1, rational sampling converters can be also implemented with other structures, e.g., various Farrow structures [19]–[22] (see also the references in these articles). The design and implementation of those converters has a very different philosophy compared to the one studied in this paper. A fair comparison between those designs and the proposed one would require plenty of study cases, especially with various values of L and M, and, as such, it is out of the scope of this paper.

The outline of this paper is as follows: Section II develops, in terms of three properties, those basic input-output relations for the rational sampling rate converter of Figure 1, which are utilized later on in order to make the overall paper easier to understand. For the sake of simplifying the discussion in this section, the coefficient symmetry of the filter H(z), which is given by (2b), is omitted, but is taken back into consideration in the rest of this contribution. Section III gives two illustrative examples on how to optimally exploit the coefficient symmetry of linear-phase FIR filters for building sampling rate converters by rational factors 2/3 and 5/3 in order to minimize the number of multipliers required in the implementation. Based on these examples and the start-up properties developed in Section II, the proposed overall implementation scheme is described in Section IV. For this purpose, first, two implementations, which can be developed in straightforward manners for appropriate combinations of M, L, and N, are generated and, then, based on these implementations, the overall scheme is developed for arbitrary combinations of M, L, and N. The implementation complexity of the proposed method is given in Section V together with four examples showing the efficiency of the proposed method over other ones existing in the literature. Finally, some concluding remarks are given in Section VI.

II. Input-Output Relations for the Rational Sample -Rate Converter Under Consideration

In this section, some basic relations for the system of Figure 1 are derived for arbitrary values of L, M, and N. These relations are used in Sections III and IV for generating an efficient implementation structure for the rational sampling-rate converter under consideration. There are three distinct parts in this section. First, the time-domain input-output relation for the system of Figure 1 is derived. Second, a compact matrix representation of the input-output relation is presented that is more suitable for generating an efficient implementation. Third, some relevant properties of the resulting matrix representation are discussed.

A. Basic Time-Domain Relations between Input and Output Samples

For the rational sampling-rate converter shown in Figure 1 the time-domain relations between the four sequences involved in this figure are expressible sequentially as

$$u[\eta] = \begin{cases} x[\eta / L] & \text{for } \eta = 0, L, 2L, ... \\ 0 & \text{otherwise} \end{cases} \tag{3a}$$

$$w[\eta] = \sum_{k=0}^{N} h_k\, u[\eta - k] \tag{3b}$$

$$y[n] = w[Mn]\,, \tag{3c}$$

where h_k's are the coefficients of the transfer function $H(z)$, as given by (2a). The coefficient symmetry, as given by (2b), is not used in this section but it will be taken back into account in Sections III and IV. Moreover, without loss of generality, it is assumed that $x[m] = 0$ for $m < 0$. The direct combination of equations (3a)–(3c) leads, after some reasoning, to the following equation:

$$y[n] = \sum_{k=0}^{N} h_k\, x\left[\frac{Mn-k}{L}\right]. \tag{4}$$

Due to the fact that the input samples $x[m]$ exist only at integer-valued time instants m, in the above summation only those terms for which $(Mn–k)/L$ is integer contribute to the output value $y[n]$, that is, the summation contains only approximately $\lfloor N/L \rfloor$ terms.

B. Input-Output Relations in Matrix Form

Based on (4), the $(n+KL)$th output sample with K being an integer, is

$$\begin{aligned} y[n+KL] &= \sum_{k=0}^{N} h_k\, x\left[\frac{M(n+KL)-k}{L}\right] \\ &= \sum_{k=0}^{N} h_k\, x\left[\frac{Mn-k}{L} + KM\right]. \end{aligned} \tag{5}$$

By comparing (4) and (5), it can be observed that when evaluating the output samples $y[n]$ and $y[n+KL]$ the same set of filter coefficients h_k is in use. This is due to the fact that the input samples $x[m]$ exist only for an integer value of m and in both cases this occurs only when $(Mn–k)/L$ is an integer. The difference is that in (5), when determining $y[n+KL]$, the input samples $x[m]$ are shifted by KM input time units in comparison to the evaluation of $y[n]$.

The above observation implies that the time-domain input-output relation of Figure 1 is uniquely characterized by the following set of L equations:

$$y[n+\ell] = \sum_{k=0}^{N} h_k\, x\left[\frac{M(n+\ell)-k}{L}\right] \quad \text{for } \ell = 0,1, ..., L-1\,, \tag{6}$$

where only those summation terms (indices) k for which $(M(n+\ell)-k)/L$ is an integer are involved in the ℓth equation. The indices k that are used for a given ℓ can be determined by rewriting $(M(n+\ell)-k)/L$ as

$$\frac{M(n+\ell)-k}{L} = \frac{Mn}{L} - \frac{k-\ell M}{L}. \tag{7}$$

In (6), the start-up value of n is not fixed and can thus be freely chosen. By selecting n to take on values 0, L, $2L$, …, the first part on the right-hand side of (7) can be written as

$$m = \frac{M}{L} n\,, \tag{8}$$

where m is also integer and takes on the values $m = 0$, M, $2M$… . Such selections of n and m considerably simplify the forthcoming discussions in this contribution. The second part on the right-hand side of (7), in turn, becomes an integer for[2]

$$k = \mathrm{mod}(\ell M, L) + K_\ell L\,, \tag{9}$$

where K_ℓ is an integer. Moreover, since in (6) only the values of k satisfying $0 \le k \le N$ are of interest, K_ℓ is limited to values $K_\ell = 0,1,..., K_\ell^{\max}$ with

$$K_\ell^{\max} = \lfloor (N - \mathrm{mod}(\ell M, L))/L \rfloor = \left\lfloor \frac{N-\ell M}{L} \right\rfloor + \left\lfloor \frac{\ell M}{L} \right\rfloor. \tag{10}$$

Here, the following identity has been applied:

$$\mathrm{mod}(\ell M, L) = \ell M - \lfloor \ell M / L \rfloor L\,. \tag{11}$$

Based on (9) and (10), (6) can be rewritten for a given ℓ by means of the summation over K_ℓ's as

$$y[n+\ell] = \sum_{K_\ell=0}^{K_\ell^{\max}} h_{\ell M + (K_\ell - \lfloor \ell M/L \rfloor) L}\, x\left[\frac{Mn}{L} - K_\ell + \left\lfloor \frac{\ell M}{L} \right\rfloor\right]. \tag{12}$$

By including the relation (8) between m and n, (12) can be expressed in a vector multiplication form as

$$y[n+\ell] = \begin{bmatrix} h_{\ell M - \lfloor \ell M/L \rfloor L} \\ h_{\ell M - \lfloor \ell M/L \rfloor L + L} \\ \vdots \\ h_{\ell M + \lfloor (N-\ell M)/L \rfloor L} \end{bmatrix}^T \begin{bmatrix} x[m + \lfloor \ell M/L \rfloor] \\ x[m + \lfloor \ell M/L \rfloor - 1] \\ \vdots \\ x[m - \lfloor (N-\ell M)/L \rfloor] \end{bmatrix} \tag{13}$$

for $\ell = 0,1, ..., L-1$.

After some further considerations, for the rational sampling converter shown in Figure 1 with the given rational factor L/M and a filter order N, L consecutive output samples, $y[n+\ell]$ for $\ell = 0,1, ..., L-1$, can be expressed by rewriting (13) in a compact matrix form as a function of the input samples in the time domain as

[2] $\mathrm{mod}(\ell, k)$ stands for modulo after division between ℓ and k, with ℓ and k being integers.

$$\mathbf{H}_{L\times(p+q+1)}=\begin{bmatrix} h_{-pL} & \cdots & h_{-L} & h_0 & h_L & \cdots & h_{(q-1)L} & h_{qL} \\ h_{M-pL} & \cdots & h_{M-L} & h_M & h_{M+L} & \cdots & h_{M+(q-1)L} & h_{M+qL} \\ h_{2M-pL} & \cdots & h_{2M-L} & h_{2M} & h_{2M+L} & \cdots & h_{2M+(q-1)L} & h_{2M+qL} \\ \cdots & & \cdots & \cdots & \cdots & & \cdots & \cdots \\ h_{(L-1)M-pL} & \cdots & h_{(L-1)M-L} & h_{(L-1)M} & h_{(L-1)M+L} & \cdots & h_{(L-1)M+(q-1)L} & h_{(L-1)M+qL} \end{bmatrix} \tag{14c}$$

$$\mathbf{y}_{n,L}=\mathbf{H}_{L\times(p+q+1)}\mathbf{x}_{m+p,m-q}\,, \tag{14a}$$

where $\mathbf{y}_{n,L}$ is a vector of the following L consecutive output samples:

$$\mathbf{y}_{n,L}=\big[y[n] \quad y[n+1] \quad y[n+2] \quad \ldots \quad y[n+L-1]\big]^T, \tag{14b}$$

$\mathbf{H}_{L\times(p+q+1)}$ is an L by $p+q+1$ matrix containing the filter coefficients as shown by (14c) at the top of the page with

$$q=\lfloor N/L \rfloor \tag{14d}$$

$$p=\lfloor (L-1)M/L \rfloor \tag{14e}$$

and

$$h_k=0 \text{ for } k<0 \text{ and } k>N. \tag{14f}$$

Finally, $\mathbf{x}_{m+p,m-q}$ is a vector containing $p+q+1$ consecutive input signal samples $x[m]$ in the descending order as follows:

$$\mathbf{x}_{m+p,m-q}=\big[x[m+p] \quad x[m+p-1] \quad \ldots \quad x[m-q]\big]^T. \tag{14g}$$

C. Properties of the Input-Output Matrix $\mathbf{H}_{L\times(p+q+1)}$

This subsection states three important properties in the relationships between the elements in the matrix $\mathbf{H}_{L\times(p+q+1)}$ that are of significant value in Section IV when deriving an efficient structure for implementing the rational L/M sampling rate converter of Figure 1 for arbitrary values of L, M, and N. Since some properties discussed in this section refer to the polyphase components of the transfer function $H(z)$, before discussing these three properties, the transfer function $H(z)$ is decomposed into L polyphase components as

$$H(z)=\sum_{\mu=0}^{L-1} z^{-\mu} H_\mu(z^L)\,, \tag{15a}$$

where

$$H_\mu(z)=\sum_{n=0}^{N_\mu} h_{\mu+nL} z^{-n}=\mathbf{h}_\mu\cdot[1\; z^{-1}\; z^{-2}\; \cdots\; z^{-q}]^T \tag{15b}$$

with

$$\mathbf{h}_\mu=[h_\mu \quad h_{\mu+L} \quad h_{\mu+2L} \quad \cdots \quad h_{\mu+N_\mu L}] \tag{15c}$$

and

$$N_\mu=\left\lfloor \frac{N-\mu}{L} \right\rfloor \tag{15d}$$

for $\mu=0,1,\ldots,L-1$.

***Property* 1**: The pth column[3] in the matrix $\mathbf{H}_{L\times(p+q+1)}$ contains the following elements:

$$\tilde{\mathbf{h}}=[h_0 \quad h_M \quad h_{2M} \quad \cdots \quad h_{(L-1)M}]^T. \tag{16}$$

This can be seen directly from (14c). Moreover, based on (8), it is straightforward to show that the elements of the vector $\tilde{\mathbf{h}}$ are those filter coefficients which multiply the single input sample $x[m]$.

***Property* 2**: The ℓth row in the matrix $\mathbf{H}_{L\times(p+q+1)}$ for $\ell=0,1,\ldots,L-1$ contains the μ_ℓth polyphase component of the transfer function $H(z)$, where

$$\mu_\ell=\mathrm{mod}(\ell M, L)\,. \tag{17}$$

Based on Property 1, the ℓth row contains the coefficient $h_{\ell M}$. On the other hand, based on Property 2 and (15c) the μ_ℓth polyphase component contains the coefficients $h_{\mathrm{mod}(\ell M,L)+rL}$ for $r=0,1,\ldots,N_{\mathrm{mod}(\ell M,L)}$. By looking only at the indices, it can be observed that there exist such $R\in\{0, 1, \ldots, N_{\mathrm{mod}(\ell M,L)}\}$ for which

$$\mathrm{mod}(\ell M, L)+RL=\ell M\,, \tag{18}$$

that is, the Rth element in the μ_ℓth polyphase component is $h_{\ell M}$. This equation, after applying the identity (11), can be rewritten as

$$R=\lfloor \ell M/L \rfloor. \tag{19}$$

The combination of the above equation and Property 1 implies that the Rth coefficient $h_{\ell M}$ is in column p. Based on this fact, it turns out that there are p–R zeros before the first nonzero coefficient in the ℓth row. Similar evaluation can be performed for determining the number of zero-valued coefficients in the ℓth row after the last coefficient from the transfer function $H(z)$ in the μ_ℓth polyphase component. Consequently, by denoting the number of zero-valued coefficients before (after) the first (last) coefficient in the μ_ℓth polyphase component with $\mu_\ell^{(b)}$ ($\mu_\ell^{(a)}$), they can be evaluated as

$$\mu_\ell^{(b)}=p-\lfloor \ell M/L \rfloor \tag{20a}$$

$$\mu_\ell^{(a)}=q-\lfloor (N-\ell M)/L \rfloor. \tag{20b}$$

[3] In this paper, rows and column are counted starting with zero.

Property 3: The matrix $\mathbf{H}_{L\times(p+q+1)}$, as given by (14c), can be expressed, in terms of the coefficients of the Nth-order transfer function $H(z)$, as

$$\mathbf{H}_{L\times(p+q+1)} = \begin{bmatrix} \mathbf{h}_0^{(H)} \\ \mathbf{h}_1^{(H)} \\ \mathbf{h}_2^{(H)} \\ \cdots \\ \mathbf{h}_\ell^{(H)} \\ \cdots \\ \mathbf{h}_{L-1}^{(H)} \end{bmatrix}, \tag{21a}$$

where

$$\mathbf{h}_\ell^{(H)} = \begin{bmatrix} \mathbf{0}_{1,\mu_\ell^{(b)}} & \mathbf{h}_{\mu_\ell} & \mathbf{0}_{1,\mu_\ell^{(a)}} \end{bmatrix} \tag{21b}$$

for $\ell = 0, 1, \ldots, L-1$. Here, $\mathbf{0}_{k,\ell}$ is a zero matrix of size k by ℓ and $\mathbf{h}_{\mu_\ell}$ is a vector containing those filter coefficients which belong to the μ_ℓ th polyphase component, as given by (15c), whereas $\mu_\ell^{(b)}$ and $\mu_\ell^{(a)}$ are given by (20a) and (20b), respectively. Consequently, the L rows of the matrix $\mathbf{H}_{L\times(p+q+1)}$ are shifted versions of the L polyphase components of the filter transfer function $H(z)$.

It should be pointed out that since $\mathrm{mod}(\ell M, L)$ does not depend on N, the order of $H(z)$, the way of distributing the L polyphase components of this transfer function among the L rows of the matrix $\mathbf{H}_{L\times(p+q+1)}$ is also independent of N. This fact simplifies the discussion in the following sections.

In the above discussions, the symmetry property (2b) has not been taken into account. After including this property, some redundancy will appear in the matrix $\mathbf{H}_{L\times(p+q+1)}$, as given by (14c). This fact will be utilized in the following sections aiming at reducing the implementation complexity of the overall system as much as possible.

III. Sampling Rate Converters with Rational Conversion Factors – Examples

The purpose of this section is to clarify, by means of two illustrative examples, the forthcoming discussion on how to efficiently exploit the coefficient symmetry of linear-phase FIR filters when using these filters for implementing a sampling rate converter shown in Figure 1.

A. Rational Sampling Rate Conversion Factor 2/3

This subsection considers a rational sampling rate converter by a factor of L/M, as shown in Figure 1, with $L = 2$, $M = 3$, and $N = 11$. In this case, the output sampling frequency is decreased by 3/2 with respect to the input sampling frequency. Based on the discussion in Section II.B and particularly (14a), the relations between L=2 consecutive output samples, $y[n]$ and $y[n+1]$, and the input samples, $x[m]$, is expressible in a matrix form as

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = \begin{bmatrix} 0 & h_0 & h_2 & h_4 & h_6 & h_8 & h_{10} \\ h_1 & h_3 & h_5 & h_7 & h_9 & h_{11} & 0 \end{bmatrix} \cdot \mathbf{x}_{m+1,m-5}, \tag{22}$$

where, according to (8) as well as the discussion related to it, $n = 0, L, 2L, \ldots = 0, 2, 4, \ldots$ and the corresponding m derived from (8) results in $m = (3/2)n = 0, 3, 6, \ldots$, whereas the vector $\mathbf{x}_{m+1,m-5}$ is defined by (14g). Taking into account the filter coefficient symmetry given by (2b), (22) can be rewritten as

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = \begin{bmatrix} 0 & h_0 & h_2 & h_4 & h_5 & h_3 & h_1 \\ h_1 & h_3 & h_5 & h_4 & h_2 & h_0 & 0 \end{bmatrix} \cdot \mathbf{x}_{m+1,m-5}. \tag{23}$$

In order to generate a form that is appropriate for an efficient implementation, the above equation is decomposed into two distinct parts as follows:[4]

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = h_4 x[m-2] \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 & h_0 & h_2 & h_5 & h_3 & h_1 \\ h_1 & h_3 & h_5 & h_2 & h_0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{m+1,m-1} \\ \mathbf{x}_{m-3,m-5} \end{bmatrix}. \tag{24}$$

The first term is very simple, thereby enabling one to implement it directly. The filter coefficient matrix in the second term is a so-called centrosymmetric matrix [17], [23]. Hence, it can be efficiently implemented by using the following decomposition:

$$\begin{bmatrix} 0 & h_0 & h_2 & h_5 & h_3 & h_1 \\ h_1 & h_3 & h_5 & h_2 & h_0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{m+1,m-1} \\ \mathbf{x}_{m-3,m-5} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_0 & c_1 & c_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & d_2 & d_1 & d_0 \end{bmatrix} \mathbf{x}_{m+1,m-5}^{(2)}, \tag{25a}$$

where

$$c_0 = h_1/2, \quad d_0 = -h_1/2,$$
$$c_1 = (h_0 + h_3)/2, \quad d_1 = (h_0 - h_3)/2, \tag{25b}$$
$$c_2 = (h_2 + h_5)/2, \quad d_2 = (h_2 - h_5)/2,$$

and

$$\mathbf{x}_{m+1,m-5}^{(2)} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{J}_3 \\ \mathbf{J}_3 & -\mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{m+1,m-1} \\ \mathbf{x}_{m-3,m-5} \end{bmatrix}. \tag{25c}$$

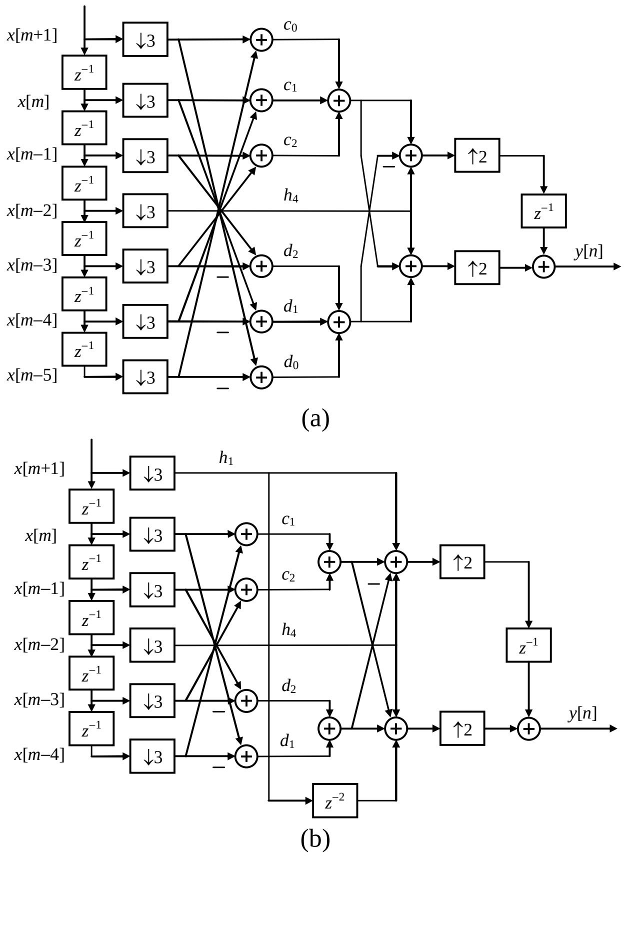
The variables c_0, c_1, c_2, d_0, d_1, and d_2 depend only on the filter coefficients and can be pre-calculated. The matrices $\mathbf{I}_3$ and $\mathbf{J}_3$ are 3 by 3 identity and counter-identity matrices, respectively. In a more general form, for $\ell < k$, the above term containing the input data can be written as

[4] The matrix decomposition in this paper is different from the one given in [18] in order to generate a more general approach for all combinations of N, L, and M.

$$\mathbf{x}_{k,\ell}^{(\lambda-1)} = \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix} \begin{bmatrix} \mathbf{x}_{k,k-\lambda+1} \\ \mathbf{x}_{\ell+\lambda-1,\ell} \end{bmatrix} = \begin{bmatrix} x[k]+x[\ell] \\ x[k-1]+x[\ell+1] \\ \cdots \\ x[k-\lambda+1]+x[\ell+\lambda-1] \\ x[k-\lambda+1]-x[\ell+\lambda-1] \\ \cdots \\ x[k-1]-x[\ell+1] \\ x[k]-x[\ell] \end{bmatrix}, \tag{26}$$

where the integer-valued parameter λ depends on N, L, and M in the manner to be discussed in Section IV.

The overall implementation structure for the example under consideration is shown in Figure 2(a). For generating two output samples, $y[n]$ and $y[n+1]$, this implementation requires 7 multipliers and 14 adders. This corresponds to 3.5 multiplications and 7 additions per output sample. The implementation of the same system without utilizing the coefficient symmetry (see, e.g., [11]) requires 6 multiplications and 5 additions per output sample. Hence, even in this simple example, the computational savings, in terms of the number of multiplications per output sample, are considerable. The downside of this implementation is a slight increase in the number of additions per output sample.



(a)

(b)

Figure 2. Implementation structure for a rational sampling rate converter by a factor of 2/3 using a linear-phase FIR filter of order $N = 11$. (a) Proposed structure. (b) Alternative structure with additional delays.

The following remark is in place: When exploiting the coefficient symmetry of an FIR filter, it is expected that the number of required multiplications per output sample is approximately halved. This is not the case for rational sampling rate converters implemented by the proposed method. As it can be seen in the above example, the achieved reduction is $6-3.5 = 2.5 < 6/2 = 3$. The reason for this lies in the fact that the coefficient with the value h_1 is not implemented efficiently. In other words, as can be seen in (25b), the variables c_0 and d_0 depend only on h_1 and not on two different coefficients as other pairs of variables c_k and d_k for $k = 1, 2$, thereby the coefficient symmetry is not utilized to its full potential. However, in this particular example, the implementation complexity can be further reduced by rewriting (24) as

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = h_1 \begin{bmatrix} x[m-5] \\ x[m+1] \end{bmatrix} + h_4 x[m-2] \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} h_0 & h_2 & h_5 & h_3 \\ h_3 & h_5 & h_2 & h_0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{m,m-1} \\ \mathbf{x}_{m-3,m-4} \end{bmatrix}. \tag{27}$$

In the above equation, the filter coefficient h_1 is implemented only once and multiplies both samples $x[m-5]$ and $x[m+1]$. By taking into account down-sampling, it is obvious that the sample $x[m-5]$ is a delayed version of the sample $x[m+1]$, with the delay being equal to two at the lowered sampling rate, that is, $m+1-(m-5) = 6 = 2M$. Therefore, by using additional delay elements in the structure, as illustrated in Figure 2(b), the overall implementation complexity reduces to 3 multiplications and 6 additions per output sample. In this case, the coefficient symmetry is completely utilized. However, it should be noted that this cannot be done for all rational sampling rate converters because the above-mentioned delay value depends on the relation between the filter order N and the down-sampling factor M and, therefore, in some cases this delay does not have an integer value, as is desired in practical implementations. Moreover, for large filter orders, such an approach would considerably increase the memory consumption of the implementation because the required number of extra delay elements is proportional to the filter order. It should be also noted that for longer filters, the non-efficient implementation of h_1 contributes proportionally less to the overall implementation complexity. Therefore, this paper concentrates on the implementation structures as the one shown in Figure 2(a), that is, structures without the extra (eventually possible) delays.

B. Rational Sampling Rate Conversion Factor 5/3

This subsection considers a rational sampling rate converter by a factor of L/M, as shown in Figure 1, with $L = 5$, $M = 3$, and $N = 23$. In this case, the output sampling frequency is increased by 5/3 with respect to the input sampling frequency. By following the discussion in Section II.B and taking into account the coefficient symmetry given by (2b), five conse-

R. Bregović, Y. J. Yu, T. Saramäki, and Y. C. Lim, "Implementation of linear-phase FIR filters for a rational sampling-rate conversion utilizing the coefficient symmetry," *IEEE Trans. Circuits Syst. I*, vol. 58, pp. 548–561, March. 2011.

quent output samples can be expressed as a function of the input samples in a matrix form as

$$\begin{bmatrix} y[n] \\ y[n+1] \\ y[n+2] \\ y[n+3] \\ y[n+4] \end{bmatrix} = \begin{bmatrix} 0 & 0 & h_0 & h_5 & h_{10} & h_8 & h_3 \\ 0 & 0 & h_3 & h_8 & h_{10} & h_5 & h_0 \\ 0 & h_1 & h_6 & h_{11} & h_7 & h_2 & 0 \\ 0 & h_4 & h_9 & h_9 & h_4 & 0 & 0 \\ h_2 & h_7 & h_{11} & h_6 & h_1 & 0 & 0 \end{bmatrix} \mathbf{x}_{m+2,m-4} \qquad (28)$$

for $n = 0, L, 2L, \ldots = 0, 5, 10, \ldots$. Here, the corresponding values of m are derived from (8), resulting in $m = (3/5)\, n = 0, 3, 6, \ldots,$ whereas the vector $\mathbf{x}_{m+2,m-4}$ is defined by (14g). In order to implement the above system efficiently, it is split into the following two parts:

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = \begin{bmatrix} h_0 & h_5 & h_{10} & h_8 & h_3 \\ h_3 & h_8 & h_{10} & h_5 & h_0 \end{bmatrix} \mathbf{x}_{m,m-4} = \\ = h_{10}\, x[m-2] \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} h_0 & h_5 & h_8 & h_3 \\ h_3 & h_8 & h_5 & h_0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{m,m-1} \\ \mathbf{x}_{m-3,m-4} \end{bmatrix} \qquad (29a)$$

$$\begin{bmatrix} y[n+2] \\ y[n+3] \\ y[n+4] \end{bmatrix} = \begin{bmatrix} 0 & h_1 & h_6 & h_{11} & h_7 & h_2 \\ 0 & h_4 & h_9 & h_9 & h_4 & 0 \\ h_2 & h_7 & h_{11} & h_6 & h_1 & 0 \end{bmatrix} \mathbf{x}_{m+2,m-3}. \qquad (29b)$$

Both of these two parts can be implemented similarly to the example in the previous subsection as

$$\begin{bmatrix} y[n] \\ y[n+1] \end{bmatrix} = h_{10}\, x[m-2] \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \\ + \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} & 0 & 0 \\ 0 & 0 & d_{01} & d_{00} \end{bmatrix} \mathbf{x}^{(1)}_{m,m-4} \qquad (30a)$$

$$\begin{bmatrix} y[n+2] \\ y[n+3] \\ y[n+4] \end{bmatrix} = \\ = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} c_{10} & c_{11} & c_{12} & 0 & 0 & 0 \\ 0 & h_4 & h_9 & 0 & 0 & 0 \\ 0 & 0 & 0 & d_{12} & d_{11} & d_{10} \end{bmatrix} \mathbf{x}^{(2)}_{m+2,m-3} \qquad (30b)$$

with

$$\begin{aligned} c_{00} &= (h_0 + h_3)/2, & d_{00} &= (h_0 - h_3)/2, \\ c_{01} &= (h_5 + h_8)/2, & d_{01} &= (h_5 - h_8)/2, \\ c_{10} &= h_2/2, & d_{10} &= -h_2/2, \\ c_{11} &= (h_1 + h_7)/2, & d_{11} &= (h_1 - h_7)/2, \\ c_{12} &= (h_6 + h_{11})/2, & d_{12} &= (h_6 - h_{11})/2. \end{aligned} \qquad (30c)$$

The structure for implementing (30a) and (30c) is shown in Figure 3.

For generating five output samples, this implementation requires 13 multipliers and 23 adders. This corresponds to 2.6 multiplications and 4.6 additions per output sample. In comparison, the polyphase implementation that does not utilize the coefficient symmetry requires 4.8 multiplications and 3.8 additions per output sample.

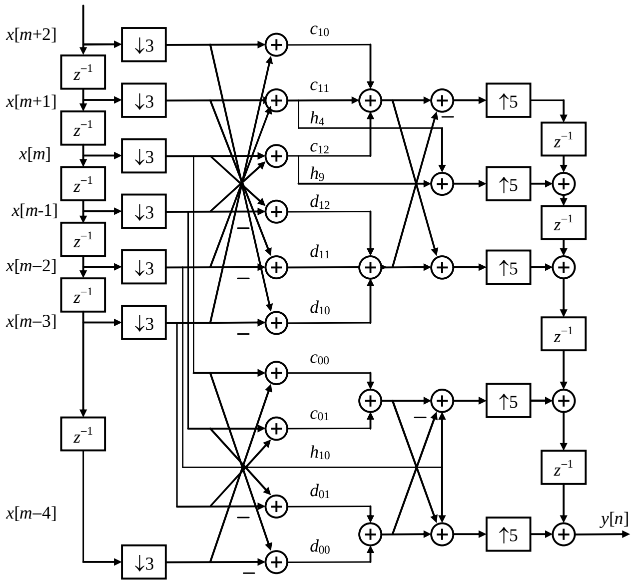


Figure 3. Proposed implementation structure for a rational sampling rate converter by a factor of 5/3 using a linear-phase FIR filter of order $N = 23$.

IV. Sampling Rate Converters with Rational Conversion Factors – Generalization

As seen from the examples in Section III, slightly different implementation structures are achieved depending on the rational factors as well as the filter orders. In this section, based on the discussion of Section II, guidelines are given on how to derive an efficient implementation structure for a given rational factor L/M and a filter order N. This effectively means to find an efficient implementation of the system given by (14a) when simultaneously taking into account the coefficient symmetry given by (2b).

The proposed method achieving this goal will be divided into three cases. The first two cases, which are referred to as Case A and Case B, respectively, consider rational sampling rate converters with special relations between the parameters N, M and L, whereas the third case, referred to as Case C, shows how a system with any combination of N, M, and L can be converted to those two special cases in such manner that an efficient implementation[5] is achieved also in this general case.

A. Case A: $N = M(L-1) + (2k+1)L$

This subsection concentrates on rational sampling rate converters, for which the filter order is given by

$$N = M(L-1) + (2k+1)L, \qquad (31)$$

where k is an integer. For these converters, the input-output relation as given by (14a)–(14g) deduces, after some straight-

[5] Examples in Sections III.A and III.B correspond to Case B and Case C, respectively.

forward manipulations, as follows. First, (14a) is expressible as

$$\mathbf{y}_{n,L} = \mathbf{H}_{L\times(p+q+1)}\mathbf{x}_{m+p,m-q} = \mathbf{H}_\mathbf{A}\mathbf{x}_{m+p,m-q}\,, \tag{32a}$$

where the size of the matrix $\mathbf{H}_\mathbf{A}$ is L_H by 2λ with

$$L_H = L \tag{32b}$$

$$\lambda = (q+p+1)/2\,. \tag{32c}$$

Second, after applying the coefficient symmetry, as given by (2b), the matrix $\mathbf{H}_\mathbf{A}$ can be separated into two matrices as follows:

$$\mathbf{H}_\mathbf{A} = [\mathbf{H}_1 \;\; \mathbf{H}_2], \tag{33a}$$

where the matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ are given by (33b) and (33c), respectively (See the top of the page). Moreover, due to the coefficient symmetry, it turns out that

$$\mathbf{H}_2 = \mathbf{J}_{L_H}\mathbf{H}_1\mathbf{J}_\lambda\,. \tag{34}$$

Finally, for later use, the elements of the matrix $\mathbf{H}_\mathbf{A}$ are denoted as $h^{(A)}_{k,\ell}$ for $k = 0, 1, ..., L_H–1$ and $\ell = 0, 1, ..., 2\lambda–1$, that is, $h^{(A)}_{k,\ell}$ is the element in the kth row and ℓth column of the matrix $\mathbf{H}_\mathbf{A}$.

Third, most importantly, the matrix $\mathbf{H}_\mathbf{A}$, as given by (33a), becomes inherently a centrosymmetric matrix [17], [23], [24]. Therefore, it can be decomposed into one of the following two forms:

$$\mathbf{H}_\mathbf{A} = \begin{bmatrix} \mathbf{I}_{\lfloor L_H/2\rfloor} & 0 & \mathbf{J}_{\lfloor L_H/2\rfloor} \\ 0 & 1 & 0 \\ \mathbf{J}_{\lfloor L_H/2\rfloor} & 0 & -\mathbf{I}_{\lfloor L_H/2\rfloor} \end{bmatrix} \times \begin{bmatrix} \mathbf{C} & \mathbf{0}_{\lfloor L_H/2\rfloor,\lambda} \\ \mathbf{e} & \mathbf{0}_{1,\lambda} \\ \mathbf{0}_{\lfloor L_H/2\rfloor,\lambda} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix} \tag{35a}$$

$$\mathbf{H}_\mathbf{A} = \begin{bmatrix} \mathbf{I}_{L_H/2} & \mathbf{J}_{L_H/2} \\ \mathbf{J}_{L_H/2} & -\mathbf{I}_{L_H/2} \end{bmatrix} \times \begin{bmatrix} \mathbf{C} & \mathbf{0}_{L_H/2,\lambda} \\ \mathbf{0}_{L_H/2,\lambda} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix}, \tag{35b}$$

depending on whether the parameter L_H is odd or even, respectively. Here, the matrices $\mathbf{C}$ and $\mathbf{D}$ and the vector $\mathbf{e}$ are expressible as

$$\mathbf{C} = \begin{bmatrix} c_{0,0} & c_{0,1} & \cdots & c_{0,\lambda-1} \\ \vdots & \vdots & & \vdots \\ c_{\lfloor L_H/2\rfloor-1,0} & c_{\lfloor L_H/2\rfloor-1,1} & \cdots & c_{\lfloor L_H/2\rfloor-1,\lambda-1} \end{bmatrix} \tag{36a}$$

$$\mathbf{D} = \begin{bmatrix} d_{\lfloor L_H/2\rfloor-1,\lambda-1} & \cdots & d_{\lfloor L_H/2\rfloor-1,1} & d_{\lfloor L_H/2\rfloor-1,0} \\ \vdots & & \vdots & \vdots \\ d_{0,\lambda-1} & \cdots & d_{0,1} & d_{0,0} \end{bmatrix} \tag{36b}$$

$$\mathbf{e} = [e_0 \quad e_1 \quad \cdots \quad e_{\lambda-1}], \tag{36c}$$

where

$$c_{k,\ell} = (h^{(A)}_{k,\ell} + h^{(A)}_{L_H-1-k,\ell})/2 \tag{36d}$$

$$d_{k,\ell} = (h^{(A)}_{k,\ell} - h^{(A)}_{L_H-1-k,\ell})/2 \tag{36e}$$

$$e_\ell = h^{(A)}_{\lfloor L_H/2\rfloor,\ell} \tag{36f}$$

for $k = 0, 1, ..., \lfloor L_H/2\rfloor - 1$ and $\ell = 0, 1, ..., \lambda–1$. These parameters depend only on the filter coefficients and can, thus, be pre-calculated.

Fourth, as was shown in Section III.A, the third factor on the right-hand side of the expressions of (35a) and (35b) can be combined with the vector of the input samples as follows:

$$\mathbf{x}^{(\lambda-1)}_{m+p,m-q} = \begin{bmatrix} \mathbf{I}_\lambda & \mathbf{J}_\lambda \\ \mathbf{J}_\lambda & -\mathbf{I}_\lambda \end{bmatrix} \mathbf{x}_{m+p,m-q}\,, \tag{37}$$

where the vector $\mathbf{x}_{m+p,m-q}$ of length p+q+1 given by (14g).

$$\mathbf{H}_1 = \begin{bmatrix} h_{-pL_H} & h_{-(p-1)L_H} & \cdots & h_0 & h_{L_H} & \cdots & h_{L_H(\lambda-p-2)} & h_{L_H(\lambda-p-1)} \\ h_{M-pL_H} & h_{M-(p-1)L_H} & \cdots & h_M & h_{M+L_H} & \cdots & h_{M+L_H(\lambda-p-2)} & h_{M+L_H(\lambda-p-1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ h_{(L_H-2)M-pL_H} & h_{(L_H-2)M-(p-1)L_H} & \cdots & h_{(L_H-2)M} & h_{(L_H-2)M-L_H} & \cdots & h_{(L_H-2)M+L_H(\lambda-p-2)} & h_{(L_H-2)M+L_H(\lambda-p-1)} \\ h_{(L_H-1)M-pL_H} & h_{(L_H-1)M-(p-1)L_H} & \cdots & h_{(L_H-1)M} & h_{(L_H-1)M-L_H} & \cdots & h_{(L_H-1)M+L_H(\lambda-p-2)} & h_{(L_H-1)M+L_H(\lambda-p-1)} \end{bmatrix} \quad (33b)$$

$$\mathbf{H}_2 = \begin{bmatrix} h_{N+(\lambda-q-1)L_H} & h_{N+(\lambda-q-2)L_H} & \cdots & h_{N-(q-p-1)L_H} & h_{N-(q-p)L_H} & \cdots & h_{N-(q-1)L_H} & h_{N-qL_H} \\ h_{N+(\lambda-q-1)L_H-M} & h_{N+(\lambda-q-2)L_H-M} & \cdots & h_{N-(q-p-1)L_H-M} & h_{N-(q-p)L_H-M} & \cdots & h_{N-qL_H-M} & h_{N-qL_H-M} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ h_{N+(\lambda-q-1)L_H-(L_H-2)M} & h_{N+(\lambda-q-2)L_H-(L_H-2)M} & \cdots & h_{N-(q-p-1)L_H-(L_H-2)M} & h_{N-(q-p)L_H-(L_H-2)M} & \cdots & h_{N-qL_H-(L_H-2)M} & h_{N-qL_H-(L_H-2)M} \\ h_{N+(\lambda-q-1)L_H-(L_H-1)M} & h_{N+(\lambda-q-2)L_H-(L_H-1)M} & \cdots & h_{N-(q-p-1)L_H-(L_H-1)M} & h_{N-(q-p)L_H-(L_H-1)M} & \cdots & h_{N-qL_H-(L_H-1)M} & h_{N-qL_H-(L_H-1)M} \end{bmatrix} \quad (33c)$$

Finally, for both converters, which are characterized by the matrix $\mathbf{H_A}$ that is given either by (35a) or (35b), an implementation structure can be derived in a manner similar to the ones shown in Figures 2(a) and 3.

B. Case B: $N = M(L-1) + 2kL$

This subsection concentrates on rational sampling rate converters, for which the filter order is given by

$$N = M(L-1) + 2kL, \quad (38)$$

where k is an integer. In a manner similar to the Case A converters, for these converters, the input-output relation, as given by (14a)–(14g), deduces, after some straightforward manipulations, as follows. First, (14a) becomes

$$\begin{aligned} \mathbf{y}_{n,L} &= \mathbf{H}_{L\times(p+q+1)}\mathbf{x}_{m+p,m-q} = \\ &= \mathbf{h_B} x_{m-\lambda+p} + \mathbf{H_B}\begin{bmatrix} \mathbf{x}_{m+p,m+p-\lambda+1} \\ \mathbf{x}_{m+p-\lambda-1,m-q} \end{bmatrix}, \end{aligned} \quad (39a)$$

where the matrix $\mathbf{H}_{L\times(p+q+1)}$, as given by (14c), is decomposed into two parts, namely, the matrix $\mathbf{H_B}$ of size L_H by 2λ and the vector $\mathbf{h_B}$ of length L_H with

$$L_H = L \quad (39b)$$

$$\lambda = \lfloor (q+p+1)/2 \rfloor. \quad (39c)$$

Second, after applying the coefficient symmetry, as given by (2b), the matrix $\mathbf{H}_B$ and the vector $\mathbf{h_B}$ take the following forms:

$$\mathbf{H}_B = [\mathbf{H_1} \;\; \mathbf{H_2}] \quad (40a)$$

$$\mathbf{h_B} = \begin{bmatrix} h_{L_H(\lambda-p)} \\ h_{M+L_H(\lambda-p)} \\ \cdots \\ h_{(L_H-2)M+L_H(\lambda-p)} \\ h_{(L_H-1)M+L_H(\lambda-p)} \end{bmatrix} = \begin{bmatrix} h_{L_H(\lambda-p)} \\ h_{M+L_H(\lambda-p)} \\ \cdots \\ h_{M+L_H(\lambda-p)} \\ h_{L_H(\lambda-p)} \end{bmatrix}, \quad (40b)$$

where the matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ are given by (33b) and (33c), respectively, and are related to each other by (34). The elements of the vector $\mathbf{h_B}$ are those ones of the λth column (middle column) of the matrix $\mathbf{H}_{L\times(p+q+1)}$,which is disregarded, due to the decomposition of (39a), in $\mathbf{H_B}$.

Because of the fact that the matrix $\mathbf{H_B}$ has a structure identical to that of the matrix $\mathbf{H_A}$, as given by (33a), it can be efficiently implemented by using the same procedure as described for the matrix $\mathbf{H_A}$ in Section IV.A. In order to utilize the symmetry properties of the vector $\mathbf{h_B}$, it can be implemented as

$$\mathbf{h_B} = \begin{bmatrix} \mathbf{I}_{\lfloor L_H/2 \rfloor} & \mathbf{0}_{\lfloor L_H/2 \rfloor,1} \\ \mathbf{0}_{1,\lfloor L_H/2 \rfloor} & 1 \\ \mathbf{J}_{\lfloor L_H/2 \rfloor} & \mathbf{0}_{\lfloor L_H/2 \rfloor,1} \end{bmatrix} \begin{bmatrix} h_{L_H\lambda-p} \\ h_{M+L_H\lambda-p} \\ \cdots \\ h_{M\lfloor L_H/2 \rfloor + L_H\lambda-p} \end{bmatrix} \quad (41a)$$

or

$$\mathbf{h_B} = \begin{bmatrix} \mathbf{I}_{L_H/2} \\ \mathbf{J}_{L_H/2} \end{bmatrix} \begin{bmatrix} h_{L_H\lambda-p} \\ h_{M+L_H\lambda-p} \\ \cdots \\ h_{M(L_H/2-1)+L_H\lambda-p} \end{bmatrix} \quad (41b)$$

depending on whether L_H is an odd or even integer, respectively.

C. Case C: Any combinations of *N*, *L*, and *M*

This subsection concentrates on those rational sampling rate converters, for which the filter orders satisfy neither (31) nor (38). In order to arrive at an efficient implementation also for these converters, it is advantageous to separate the input-output transfer matrix $\mathbf{H}_{L\times(p+q+1)}$, as given by (14c)–(14f), into two matrices $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ as follows:

R. Bregović, Y. J. Yu, T. Saramäki, and Y. C. Lim, “Implementation of linear-phase FIR filters for a rational sampling-rate conversion utilizing the coefficient symmetry,” *IEEE Trans. Circuits Syst. I*, vol. 58, pp. 548–561, March. 2011.

$$\mathbf{H}_{L\times(p+q+1)} = \left[\begin{array}{cc} \mathbf{0}_{L_1,p-p_1} & \mathbf{H}_{C_1} \\ \mathbf{H}_{C_2} & \mathbf{0}_{L_2,q-q_2} \end{array}\right], \qquad (42)$$

where the matrices $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ are of sizes L_1 by $q+p_1+1$ and L_2 by q_2+p+1, respectively. The corresponding two input-output relations can then be expressed as

$$\mathbf{y}_{n,L_1} = \mathbf{H}_{\mathbf{C}_1}\mathbf{x}_{m+p_1,m-q} \qquad (43a)$$

$$\mathbf{y}_{n+L_1,L_2} = \mathbf{H}_{\mathbf{C}_2}\mathbf{x}_{m+p,m-q_2}\,. \qquad (43b)$$

When separating the matrix $\mathbf{H}_{L\times(p+q+1)}$, as shown by (42), the ultimate goal is to select the matrices $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ in such a way that both of them have structures similar to those ones developed in Sections IV.A and IV.B for Case A and Case B converters, respectively. This will enable one to efficiently implement both $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ as illustrated by the example in Section III.B. As a matter of fact, it has been experimentally observed that for any input-output matrix $\mathbf{H}_{L\times(p+q+1)}$ of a rational sampling rate converter, there does exist such a representation through $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$.[6]

In order to generate the desired matrices $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$, the parameters L_1, L_2, p_1, and q_2 have to be appropriately determined. For this purpose, it is observed that due to the structure of $\mathbf{H}_{L\times(p+q+1)}$ and the properties of its polyphase terms, $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ take on the desired form provided that the last row in $\mathbf{H}_{\mathbf{C}_1}$ contains the same coefficients as the zeroth row but in the time-reversed order. As can be seen from (17), (21a), and (21b), the zeroth row of the matrix $\mathbf{H}_{L\times(p+q+1)}$ always contains the zeroth polyphase component, that is, $\mathbf{h}_0$. Moreover, as seen from (15c), due to the linear-phase property of the FIR filter under consideration, the last element in $\mathbf{h}_0$ is $h_{qL} = h_{N-qL}$ and the first element in the μth polyphase component is h_μ. Therefore, the goal is to find the rth row, that is simultaneously the last row of the matrix $\mathbf{H}_{\mathbf{C}_1}$, which contains the $\mu_r = \mu_{N-qL}$ th polyphase component. This corresponds to finding the integer $r \in \{0, 1, \ldots, L-1\}$ such that the following equation

$$\mathrm{mod}(rM, L) = N - qL \qquad (44)$$

is valid. This unique value of r can be determined by, first, evaluating $\mu_\ell = \mathrm{mod}(\ell M, L)$ for $\ell = 0, 1, \ldots, L-1$ and, then, selecting among the resulting L pairs of the parameters ℓ and μ_ℓ, the one, for which μ_ℓ becomes equal to $N-qL$. After finding r, the required remaining parameters L_1, L_2, p_1, and q_2 can then be evaluated as

$$L_1 = r + 1 \qquad (45a)$$

$$L_2 = L - L_1 \qquad (45b)$$

$$p_1 = \frac{(L_1 - 1)M - (N - qL)}{L} \qquad (45c)$$

$$q_2 = \left\lfloor \frac{N - \mathrm{mod}(L_1 M, L)}{L} \right\rfloor - \left\lfloor \frac{L_1 M}{L} \right\rfloor. \qquad (45d)$$

It should be noted that when implementing the matrices $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ as Case A or Case B matrices, the sizes of the matrices to be actually implemented are L_H by 2λ with L_H and λ defined as

$$L_H = \begin{cases} L_1 & \text{for } H_{C_1} \\ L_2 & \text{for } H_{C_2} \end{cases} \qquad (46a)$$

$$\lambda = \begin{cases} \lfloor (q + p_1 + 1)/2 \rfloor & \text{for } H_{C_1} \\ \lfloor (q_2 + p + 1)/2 \rfloor & \text{for } H_{C_2}. \end{cases} \qquad (46b)$$

In the above equation, λ is expressed in the form $\lambda = \lfloor t/2 \rfloor$ with t being an integer. For t even and odd, the corresponding matrix, which can be either $\mathbf{H}_{\mathbf{C}_1}$ or $\mathbf{H}_{\mathbf{C}_2}$, is a Case A and Case B matrix, respectively.

In order to illustrate the procedure for matrix decomposition described in this section, the example given in Section III.B will be reused. In this example, $L = 5$, $M = 3$, and $N = 23$. According to (15c), the last element in zeroth row in the matrix $\mathbf{H}_{L\times(p+q+1)}$ is $h_{qL} = h_{N-qL} = h_{23-4\bullet 5} = h_3$. Therefore,

$$h_3 \Rightarrow \mu_r = 3 \overset{(44)}{\Rightarrow} r = 1 \overset{(45a)-(45d)}{\Rightarrow} \\ \Rightarrow L_1 = 2,\ L_2 = 3,\ p_1 = 0,\ q_2 = 3. \qquad (47)$$

This can be verified by (29a) and (29b). Finally, in this example, $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ are implemented as Case B and Case A matrices, respectively.

V. IMPLEMENTATION COMPLEXITY ESTIMATION

In this section, an estimation for the implementation complexity of rational sampling rate converters realized by the proposed method is given together with four examples showing the efficiency of the proposed implementation. First, the implementation complexities are estimated for Cases A and B converters and, then, based on these complexity estimates, it is explained how to estimate the complexity for Case C converters. In all cases, C_p^* and C_p^+ stand for the number of multiplications and the number of additions per output sample, respectively.

[6] The existence of the desired separation for a general case is based only on extensive simulations, which did not come up with any counter example. However, no theoretical proof has been found up to now for validating this claim.

The implementation complexity of the matrix $\mathbf{H_A}$ of size L_H by 2λ can be estimated by

$$C_p^* = C_A^* = \lambda \tag{48a}$$

$$C_p^+ = C_A^+ = \begin{cases} \lambda + \dfrac{2\lambda}{L_H} - \dfrac{\mathrm{mod}(L_H,2)}{L_H} & \text{for} \quad L_H > 1 \\ 2\lambda - 1 & \text{for} \quad L_H = 1, \end{cases} \tag{48b}$$

where λ is either given by (32c) or (46b) and L_H is either equal to L or given by (46a) for Case A or Case C converters, respectively.

The implementation complexities of the matrix $\mathbf{H_B}$ of size L_H by 2λ and the vector $\mathbf{h_B}$ of length L_H can be estimated by

$$C_p^* = C_B^* = \lambda + \frac{1}{L_H}\left\lfloor \frac{L_H}{2} \right\rfloor \tag{49a}$$

$$C_p^+ = C_B^+ = \begin{cases} \lambda + \dfrac{2\lambda}{L_H} + 1 - \dfrac{\mathrm{mod}(L_H,2)}{L_H} & \text{for} \quad L_H > 1 \\ 2\lambda & \text{for} \quad L_H = 1, \end{cases} \tag{49b}$$

where λ is either given by (39c) or (46b) and L_H is either equal to L or given by (46a) for Case B or Case C converters, respectively.

The implementation complexity for Case C converters can thus be evaluated as

$$C_p^* = \frac{L_1 C^*_{\mathbf{H}_{C1}} + L_2 C^*_{\mathbf{H}_{C2}}}{L} \tag{50a}$$

$$C_p^+ = \frac{L_1 C^+_{\mathbf{H}_{C1}} + L_2 C^+_{\mathbf{H}_{C2}}}{L}, \tag{50b}$$

where the implementation complexities of $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ are evaluated by (48a) and (48b); and (49a) and (49b), respectively. In order to evaluate the complexity of Case C converters, first, the matrix separation has to be performed according to the discussion of Section IV.C and, then, depending on the resulting separation, the complexities of the matrices $\mathbf{H}_{\mathbf{C}_1}$ and $\mathbf{H}_{\mathbf{C}_2}$ are accordingly evaluated.

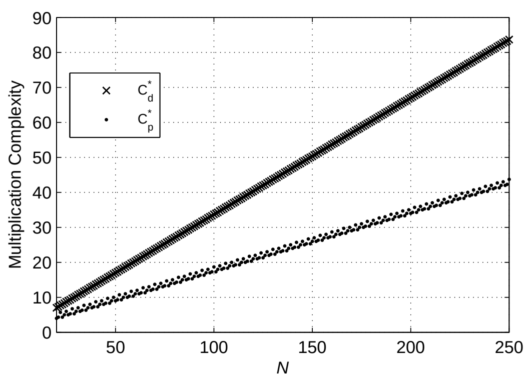
A. Rational Sampling Rate Conversion Factor 3/5

In this example, rational sampling rate converters with factor 3/5 are considered. Filters of various order have been implemented by using the proposed method and the polyphase one [11]. The corresponding implementation complexities are shown in Figure 4 and Table I. As seen in this table and figure, the proposed method results always in a system with a lower implementation complexity compared with the polyphase implementation. Moreover, this comparison shows how the complexity of the proposed implementation reduces with respect to the corresponding polyphase one when the filter order is increased. For example, as seen in Table I, for the filter order $N = 23$, the number of multiplications required by the proposed implementation is only 54.2% of that needed by the polyphase implementation that does not exploit the coefficient symmetry, whereas for $N = 212$, this figure drops down to 50.7%. As already discussed before, due to the structure of the matrix $\mathbf{H}_{L\times(p+q+1)}$, except in some special cases (see Section III.A), the number of multiplications cannot be halved in comparison with the polyphase implementation as could be expected for a linear phase FIR filter. Nevertheless, as seen in Figure 4(c), the proposed implementation approaches this desired goal as the filter order increases.

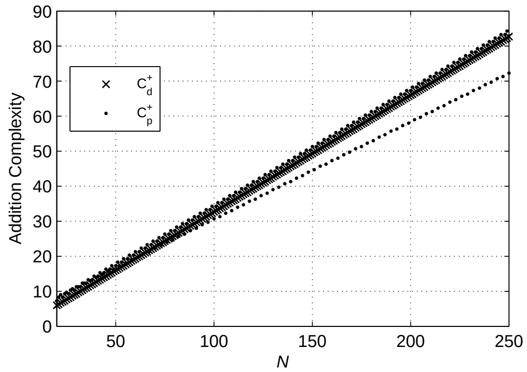
Furthermore, as seen in Table I and Figure 4, there are slight variations in the implementation complexities for sequential filter orders. This is due to the fact that for some filter orders, a more efficient matrix separation exists in comparison with others. It should also be pointed out that for this example this enables some good compromise solutions between the multiplication and addition complexity. As seen especially in Table I, by choosing a little bit different filter order, the multiplication complexity can be slightly decreased at the expense of a minor increase in the addition complexity and vice versa.

TABLE I IMPLEMENTATION COMPLEXITY FOR RATIONAL SAMPLING RATE CONVERTERS BY 3/5 IN NUMBER OF MULTIPLICATIONS (C^*) AND ADDITIONS (C^+) PER OUTPUT SAMPLE FOR THE PROPOSED AND POLYPHASE IMPLEMENTATION

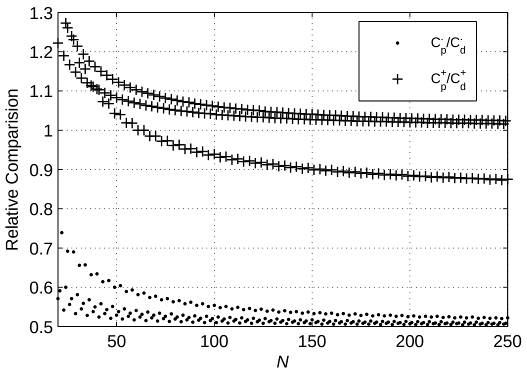
		Polyphase		Proposed		Comparison	
N	Case	C_d^*	C_d^+	C_p^*	C_p^+	C_p^*/C_d^*	C_p^+/C_d^+
23	C	8	7	4.3	8.3	0.542	1.190
209	C	70	69	35.3	70.3	0.505	1.019
210	C	70.3	69.3	36	71.3	0.512	1.029
211	A	70.7	69.7	37	61.3	0.524	0.880
212	C	71	70	36	71.3	0.507	1.019
213	C	71.3	70.3	36.3	72.3	0.509	1.028
214	B	71.7	70.7	37.7	62.3	0.526	0.882



(a)

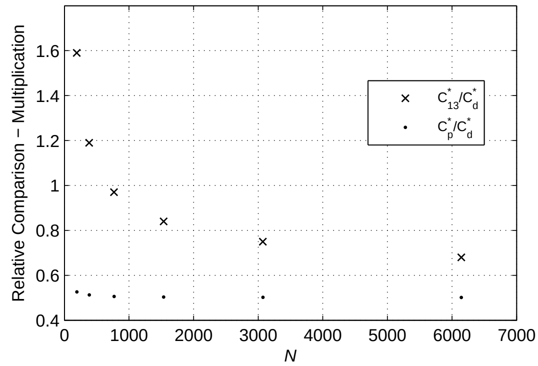


(b)

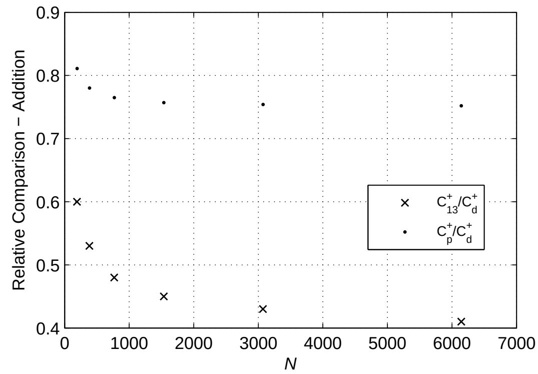


(c)

Figure 4. Implementation complexity for rational sampling rate converters by 3/5. Subindexes *d* and *p* stand for the polyphase and the proposed implementation, respectively. (a) Number of multiplications ($C*$) per output sample. (b) Number of additions (C^+) per output sample. (c) Relative comparison between the proposed and polyphase implementation.



(a)



(b)

Figure 5. Comparison between the proposed implementation and the implementation described in [13] for rational sampling rate converters by 4/3. Subindexes *p*, *d*, and 13 stand for the proposed implementation, polyphase implementation, and implementation proposed in [13], respectively. (a) Multiplication complexity. (b) Addition complexity.

B. Rational Sampling Rate Conversion Factor 4/3

In this example, rational sampling rate converters with factor 4/3 are considered. Filters of various order have been implemented with the proposed method and method introduced in [13]. The resulting comparison is shown in Figure 5. Since in [13] the results are given as a relative complexity with respect to the polyphase implementation, in order to be able to reuse the data from Table 1 in [13], in this paper, both methods are compared with respect to the polyphase implementation. As seen from the figure, the proposed method requires fewer multiplications but more additions than the method in [13]. However, the higher amount of additions is not a problem as multiplications are more costly to implement than additions. Figure 5(a) also shows that the difference between these two methods becomes smaller as the filter order increases. Nevertheless, the proposed method results always in an implementation with a smaller number of multiplications than the method in [13].

C. Rational Sampling Rate Converters with *L*th-Band Filters

In many cases, when implementing a sampling rate converter, a special type of linear-phase FIR filters is used, known as *L*th-band filters. Among many interesting properties of *L*th-band filters the following two are relevant when using such filters for building rational sampling rate converters: First, the cut-off frequency of an *L*th-band filter is located, in terms of the angular frequency, approximately at π/L, and, second, the value of the center coefficient is[7] $1/L$, whereas every *L*th impulse-response value to the left and right from the center coefficient is equal to zero. This is illustrated in Figure 6 for a third-band filter of order $N = 14$.

[7] After compensating for up-sampling by *L*, the center impulse-response coefficient becomes one.

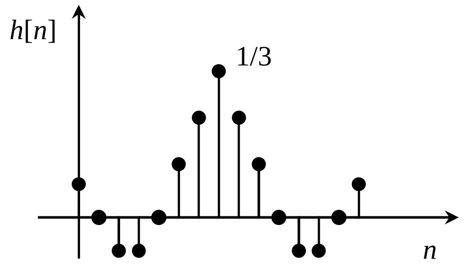


Figure 6. Impulse response of a 3rd-band filter of order $N = 14$.

When using this filter in a rational sampling rate converter by factor of 3/2, it is beneficially to utilize the above-mentioned time-domain properties in the implementation. It turns out that this is very straightforward when implementing the filter with the proposed method. That is, this method leads to the following input-output relationship:

$$\mathbf{y}_{n,2} = \begin{bmatrix} 0 & h_0 & h_3 & h_6 & h_5 & h_2 \\ 0 & h_2 & h_5 & h_6 & h_3 & h_0 \\ h_1 & h_4 & h_7 & h_4 & h_1 & 0 \end{bmatrix} \mathbf{x}_{m+1,m-4} \qquad (51a)$$

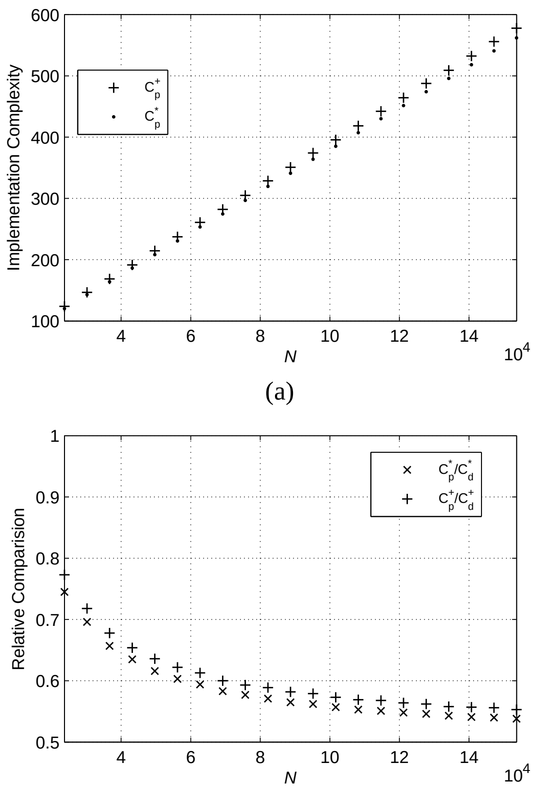
which reduces after utilizing the third-band property, that is, $h_1 = h_4 = 0$ and $h_7 = 1$, to the following simplified form:

$$\mathbf{y}_{n,2} = \begin{bmatrix} h_0 & h_3 & h_6 & h_5 & h_2 \\ h_2 & h_5 & h_6 & h_3 & h_0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}_{m,m-4}. \qquad (51b)$$

This implementation requires only 1.67 multiplications and 3.33 additions per output sample. The implementation of the same system by using the polyphase implementation [11] requires 3.33 multiplications and 2.33 additions per output sample. Without taking into account the above-mentioned properties of the Lth-band filter, the complexity would be 2.67 and 5 multiplications per output sample for the proposed and the polyphase implementation, respectively.

D. Rational Sampling Rate Conversion Factor 147/160

In order to demonstrate the performance of the method for larger values of L and M, this section concentrates on the classical example of rational sampling rate conversion between the sampling rates of 48 kHz used in a digital audio tape and 44.1 kHz used in a compact disc. This corresponds to the rational sampling rate conversion factor 147/160. Filters of various orders have been implemented by the proposed and polyphase method. The corresponding implementation complexities for the proposed method are shown in Figure 7(a). A comparison with the polyphase method is given in Figure 7(b). Based on these figures, two observations can be made. First, also for this choice of M and L, the proposed method is more efficient that the polyphase one. Second, a rational sampling rate converter given in Figure 1 has a limited usability for large values of M and L. The problem is that in this case FIR filters of a very high order are required that are difficult to design and implement. This problem can be overcome by using multistage rational sampling rate conversion, as shown e.g. in [21] with the proposed method used for implementing every stage. Alternatively, for high values of M and L, other techniques for rational sampling rate conversion, (see,e.g., [22]), can be considered.



(a)

(b)

Figure 7. Implementation complexity for rational sampling rate converters by 147/160. Subindexes d and p stand for the polyphase and the proposed implementation, respectively. (a) Number of multiplications (C^*) and additions (C^+) per output sample. (b) Relative comparison between the proposed and polyphase implementation.

VI. Concluding Remarks

This paper has proposed an efficient implementation of linear-phase FIR filters of an arbitrary filter order N for providing a sampling rate conversion by an arbitrary rational factor of L/M. In this implementation, the following three facts have been exploited: First, the filter impulse response is symmetrical, second, only every Lth filter input sample has a non-zero value, and, third, only every Mth filter output sample has to be evaluated. By utilizing these properties, an implementation that is much more efficient than the ones proposed so far has been achieved.

The following remarks regarding the proposed method should be made. First, except in some special cases, the proposed method does not reduce the number of required multiplications approximately by a factor of two compared to the polyphase implementation as it would be expected regarding the fact that only approximately half of the filter coefficients have different values. However, as seen from the examples, the proposed implementation approaches this goal as the filter order increases, thereby making the proposed approach more efficient for filters with high orders. Furthermore, as shown in Section III.A, in some cases, this goal can be achieved by adding extra delay elements.

Second, the proposed method can be straightforwardly applied to multistage rational sampling rate converters.

Third, in the case of hardware implementation, the methods presented in this paper could be combined with the theory about constant matrix multiplications [25].

Fourth, as long as the FIR filter used in the rational sampling rate converter can be designed, the proposed method can be used to implement the converter in a more efficient way than the corresponding polyphase implementation for any combination of values M, L and N. However, it should be mentioned that for large values of M and L, the proposed method as well as the polyphase one might get impractical due to high hardware requirements (number of delays, number of multipliers, number of required bits, etc.). In cases when both M and L can be factorized, this problem can be overcome by using multistage rational sampling rate convertors.

Fifth, it has been experimentally observed that in most cases, for a given L and M, it is beneficially to choose such filter order that will divide the input-output matrix into two parts of similar size (Case C convertor). In this case, due to a small redundancy in matrices $\mathbf{H_A}$ and/or $\mathbf{H_B}$ a system with smallest multiplication complexity will be obtained compared to adjacent filter orders.

VII. References

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

[2] N. J. Fliege, *Multirate Digital Signal Processing*. Chicester: John Wiley and Sons, 1994.

[3] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[4] G. Jovanovic-Dolecek, *Multirate Systems: Design and Applications*. London: Idea Group Publishing, 2002.

[5] F. Harris, *Multirate Signal Processing for Communication Systems*. Upper Saddle River, NJ: Prentice-Hall, 2004.

[6] M. Laddomada, “Design of multistage decimation filters using cyclotomic polynomials: Optimization and design issues,” *IEEE Trans. Circuits Syst. I*, vol. 55, pp. 1977–1987, July 2007.

[7] M. Laddomada, “Generalized comb decimation filters for ΣΔ A/D converters: Analysis and design,” *IEEE Trans. Circuits Syst. I*, vol. 55, pp. 994–1005, May 2007.

[8] G. J. Dolecek and M. Laddomada, “An economical class of droop-compensated generalized comb filters: Analysis and design,” *IEEE Trans. Circuits Syst. II*, vol. 57, pp. 275–279, April 2010.

[9] 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 Willey & Sons, NY, 1993, pp. 155–336.

[10] R. E. Crochiere, L. R. Rabiner, “Interpolation and decimation of digital signals – A tutorial review,” in *Proc. IEEE*, vol. 69, pp. 300–331, March 1981.

[11] C.-C. Hsiao, “Polyphase filter matrix for rational sampling rate conversion,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, April 1987, pp. 2173–2176.

[12] D. W. Barker, “Efficient Resampling Implementations,” *IEEE Signal Process. Magazine*, July 2008, pp. 114–117.

[13] H. Murukami, “Block sampling rate conversion system using real-valued fast cyclic convolution algorithms,” *IEEE Trans. Signal Process.*, April 1997, pp. 1070–1075.

[14] G. Bi, F. P. Coakley, and B. G. Evans, “Rational sampling rate conversion structures with minimum delay requirements,” *IEE Proceedings-E*, Nov. 1992, pp. 477–485.

[15] H. G. Göckler, G. Evangelista, and A. Groth, “Minimal block processing approach to fractional sample rate conversion,” *Signal Process.*, April 2001, pp. 673–691.

[16] Y. J. Yu, Y. C. Lim, and T. Saramäki, “Restoring coefficient symmetry in polyphase implementation of linear-phase filters,” *Circuits, Systems, Signal Processing*, no. 2, pp. 253–264, 2006.

[17] Z.-J. Mou “Symmetry exploitation in digital interpolators/decimators,” *IEEE Trans. Signal Process.*, vol 44, pp. 2611–2615, Oct. 1996.

[18] R. Bregović, T. Saramäki, Y. J. Yu, and Y. C. Lim, “An efficient implementation of linear-phase FIR filters for a rational sampling rate conversion,” in *Proc. IEEE Int. Symp. Circuits Syst.*, Island of Kos, Greece, May 2006, pp. 5395–5398.

[19] J. Vesma and T. Saramäki, “Polynomial-based interpolation filters – Part I: Filter synthesis,” *Circuits, Systems, and Signal Processing*, vol. 26, no. 2, pp. 115–146, 2007.

[20] D. Babić, J. Vesma, T. Saramäki, and M. Renfors, “Implementation of the transposed Farrow structure,” in *Proc. IEEE Int. Symp. Circ. Syst.*, May 2002, vol. IV, pp. 5–8.

[21] M. M. Kashtiban, S. Farazi, and M. G. Shayesteh, “Optimum structures for sample rate conversion from CD to DAT and DAT to CD using multistage interpolation and decimation,” in *Proc. 2006 IEEE Int. Symp. on Signal Process. and Information Technology*, Vancouver, Canada, Aug. 2006, pp. 633–637.

[22] K. Rajamani, Y.S. Lai, and C. W. Farrow, “An efficient algorithm for sample rate conversion from CD to DAT,” *IEEE Signal Process. Letters*, Oct. 2000, pp. 288–290.

[23] L. Datta and S. D. Morgan, “On the reducibility of centrosymmetric matrices – Applications in engineering problems,” *Circuits, Systems, and Signal Processing*, no. 1, pp. 71–96, 1989.

[24] R. Bregović, Y. J. Yu, A. Viholainen, and Y. C. Lim, “Implementation of linear-phase FIR nearly perfect-reconstruction cosine-modulated filterbanks utilizing the coefficient symmetry,” *IEEE Trans. Circuits Syst. I*, vol. 57, pp. 139–151, Jan. 2010.

[25] O. Gustafsson and H. Johansson, “Efficient implementation of FIR filter based rational sampling rate converters using constant matrix multiplication,” in *Proc. 40th Asilomar Conf. Signals, Syst., Computers*, Oct. 2006, pp. 888–891.