Minimal Block Processing Approach to Fractional Sample Rate Conversion

Heinz G. Göckler, Gennaro Evangelista, Alexandra Groth
Digital Signal Processing Group
Ruhr-Universität Bochum
D-44780 Bochum, Germany
Tel.: +49 234 32 22869
Fax: +49 234 32 14100
email: goeckler@nt.ruhr-uni-bochum.de

Abstract
The problem of synchronous Fractional Sample Rate Conversion (FSRC) of a digital signal by $L/M$, where $L$ and $M$ are coprime integers [3,13], is revisited. Based on a novel approach two different efficient causal block implementations of FSRC are concurrently derived and compared with each other. While the computational load of both structures, being performed in an LTI MIMO subsystem at the subnyquist rate $F_o/L = F_i/M$, is identical, their group delay is always different. By column and row shifts of the matrix representation of the MIMO subsystem it is possible to transform each structure into any arbitrary implementation with changed group delay. Moreover it is shown that, by structural manipulation of the signal flow graph of the MIMO subsystem, both implementations ultimately require the same amount of computation and storage in spite of different group delay. Finally, by using Nyquist($L$)filters, the maximum number of input samples to FSRC is retained at its output.

Zusammenfassung

Keywords
Fractional Sample Rate alteration, Minimal realisation, Block processing, Subnyquist processing, FIR-System


I. Introduction

In digitally implemented systems where sampled signals of various, generally to be adapted bandwidths are processed, sample rate conversion is omnipresent. One particular and yet important case of this need of sample rate alteration is the conversion of the sampling rate by a noninteger factor $L/M$, where $L$ and $M$ are positive and relatively prime integers. The concern of this paper is a novel rigorous derivation of an algorithm and the implementation of this type of sample rate conversion based on block processing with minimum overall expenditure.

The system theoretic approach to fractional sample rate alteration is characterised by the cascade connection of an interpolator for $L$-fold rate increase followed by a decimator for $M$-fold rate reduction [1,2]. The corresponding block structure is depicted in Fig.1, where the antiimaging filter of the interpolator and the antialiasing filter of the decimator have merged in one filter $H(z)$. As shown in Fig.1, all filter operations have to be performed at the highest rate $F = 1/T$ ($T$: Sampling period), which is related to the system input rate $F_i$ or output rate $F_o$, respectively, in a fixed synchronous manner by

$$F = L F_i = M F_o$$

in compliance with the objective of the fractional sample rate converter:

$$\frac{F_o}{F_i} = \frac{L}{M}$$

In Fig.1, the $z$-domain representations of the input and output signals are related to the respective input and output sampling periods (and rates) [1,2] according to

$$z_i = e^{s T_i} = e^{s T L} = z^L \quad \text{and} \quad z_o = e^{s T_o} = e^{s T M} = z^M$$

where use is made of (1).

By suitably decomposing the highly inefficient structure of Fig.1 into polyphase components, it is possible to remove all its inherent multiplications with zeros and to perform all arithmetic operations (multiplications and additions) at the subnyquist rate

$$F_s = \frac{F}{LM} = \frac{F_i}{M} = \frac{F_o}{L}$$

corresponding to the $z$-domain variable:

$$z_s = e^{s TLM} = z^{LM} = z^M = z^L$$

A somewhat heuristic derivation of this approach is given in [1,2,12]. However, the proposed polyphase structures unfortunately require a multitude of input or output commutators, respectively. These many input or output commutators, respectively, cannot immediately be replaced by one only, since they are interconnected by delay elements being operated at the input or output rate, respectively. Nevertheless, this extra expenditure is superfluous, and it will be shown in this paper how it can be avoided.
A polyphase structure of a fractional sample rate converter that overcomes the above hardware deficiency was revealed in a US-patent [3] and published in three conference papers [4,13,17], respectively. This approach requires one input commutator for $M$-fold decimation, and one output commutator for $L$-fold sample interleaving. All operations (including any memory shift) are performed at the subnyquist rate $F_s$. In [3,4,13,17], however, a proper mathematical derivation and, in particular, an exhaustive discussion of the well-defined minimal polyphase structure is missing.

In this paper, two novel systematic and rigorous derivations of two possible optimum polyphase implementations of the above defined fractional sample rate conversion (FSRC) are developed concurrently. The features and merits of both approaches applying block processing are discussed and compared with each other. The conditions for the minimal implementation are given. Finally, the results are illustrated by an example, and it is shown that both approaches can be implemented with exactly the same amount of hardware.

II. The Minimal Polyphase Implementation

The derivation of the minimal polyphase implementation of the FSRC starts from Fig.1. Let

$$H(z) = z^{-K} H_c(z), \quad K \in \mathbb{N}_0$$

where $H_c(z)$ represents a causal, possibly minimum phase FIR or IIR filter, respectively, suitably specified and designed for FSRC [1,2]. Due to the extra memory, $H(z)$ is not minimum phase but always causal and, hence, realisable. In course of the polyphase decompositions to follow, the respective minimum values of the quantity $K \in \mathbb{N}_0 = \{0,1,2,\ldots\}$ will be determined.

A. Derivation of two Optimum Structures

The derivation of the two optimum FSRC algorithms is subdivided into five logical steps. In Step 1, the FSRC filter $H_c(z)$ is decomposed into polyphase components, where subsequently always type 1 polyphase filter decomposition [1] is applied. In Step 2, the order of the respective polyphase components is systematically rearranged, such that the noble identities [1] can readily be exploited to the highest advantage in Step 3. In Step 4, in the end, each original polyphase component is subjected to a second polyphase decomposition yielding, after all, the two optimum FSRC structures: The serial input signal is blocked into a length-$M$ vector-valued signal, which is the input to a multiple-input multiple-output (MIMO) linear time-invariant (LTI) system producing a length-$L$ vector-valued signal at its output that, sub
sequently, is unblocked to form the output signal in serial form [5,6]. Finally, equivalent optimum polyphase implementations, where the (un-)blocking circuits are replaced by commutator models, are presented in Step 5.

**Step 1: Polyphase Decomposition** [1,2,4,6,7,17] of $H_c(z)$
In the outlined derivation procedure two cases must be distinguished. Define **Case a** by the $L$-branch polyphase decomposition of the FSRC filter $H_c(z)$, and set $K = \lambda$. Note that, in this case, the decomposed filter represents the input interpolator of FSRC according to Fig.1. Define **Case b** by the $M$-branch polyphase decomposition of the FSRC filter $H_c(z)$, and set $K = \mu$.
Here, in contrast to the former case, the decomposed filter represents the output decimator of Fig.1. Obviously, polyphase decomposition is straightforward for FIR filters [1,2] whereas, in general, a standard rational transfer function of an IIR filter must be subjected to an equivalence transformation before polyphase decomposition can be applied (cf. e.g. [4,7]):

$$H(z) = z^{-\lambda} \sum_{l=0}^{L-1} z^{-l} H_l(z^L)$$  \hspace{1cm} (6a)

$$H(z) = z^{-\mu} \sum_{m=0}^{M-1} z^{-m} H_m(z^M)$$  \hspace{1cm} (6b)

**Step 2: Systematic Rearrangement of Sequential Order of Polyphase Components** (6a,b)
In order to prepare the following step, the application of the noble identities [1] to reduce the overall computational burden, the first polyphase decompositions (6a,b) are rearranged by index mapping of the summation indices $l$ and $m$, respectively, on the basis of modulo arithmetic.

For **Case a** we introduce the substitution

$$l := (lM)_L = (lM) \text{ modulo } L = lM - p_l L$$  \hspace{1cm} (7a)

where $l = 0,1,...,L-1$, $p_l \in \mathbb{N}_0$ and obviously $p_0 = 0 \leq p_1 \leq ... \leq p_l \leq ... \leq p_{L-1}$. This substitution represents a one-to-one mapping according to

$$(lM)_L \in \{0,1,...,L-1\} \Leftrightarrow l \in \{0,1,...,L-1\}$$

provided that $L$ and $M$ are coprime, as assumed from the very beginning. The upper and lower bounds of $p_{L-1}$ are readily deduced from (7a) using the fact that the original index $l = 0$ is always related to $p_0 = 0$, limiting the remaining mapped indices as follows

$$1 \leq (lM)_L \leq L - 1, \quad \forall l = 1,2,...,L-1.$$

Introducing $l = L-1$ into these two inequalities yields, in conjunction with the right hand side of (7a), the interval

$$\frac{(L-1)(M-1)}{L} \leq p_{L-1} \leq \frac{(L-1)M-1}{L}$$  \hspace{1cm} (8a)

or the upper bound of $p_{L-1} \in \mathbb{N}$, respectively:
\[ p_{L-1} = \left\lfloor \frac{(L-1)M-1}{L} \right\rfloor = M - \left\lceil \frac{M+1}{L} \right\rceil \]  
(9a)

where \( 0 \leq x - \lfloor x \rfloor < 1 \) and \( 0 \leq \lfloor x \rfloor - x < 1 \). By introducing the index mapping (7a) into (6a),
and by setting
\[ \lambda = p_{L-1} L \]
(10a)
under consideration of (9a), we get
\[ H(z) = z^{-p_{L-1}L} \sum_{l=0}^{L-1} z^{-(lM-p_lL)} H_{(lM)_L}\left(z^L\right) = \sum_{l=0}^{L-1} z^{-lM} E_l(z^L) \]  
(11a)
where the quantities
\[ E_l(z^L) = H_{(lM-p_lL)}\left(z^L\right) \cdot z^{-(p_{L-1} - p_l)L} \]
(12a)
represent the \( L \) reordered polyphase components of the FSRC input interpolator augmented by some extra delay. The decomposed filter \( H(z) \) imbedded in-between an input upsampler and an output downsampler is depicted in Fig.2a.

Fig.2: Rearranged polyphase decomposition according to step 2  
Case a: \( L \)-branch polyphase decomposition  
Case b: \( M \)-branch polyphase decomposition

It is observed from (12a) in comparison with (5) and (6a) that all rearranged polyphase components are causal and, hence, immediately realisable. This is accomplished by the additional memory introduced in (5) and (6a) with \( K = \lambda \) according to (9a) and (10a). This choice of \( \lambda \) also guarantees that the extra delay is minimum. This follows from (12a) for \( l = L-1 \) since, for this index, the rearranged and original polyphase components are identical and, hence, of minimum order. Obviously (cf. Fig.2a), all components of (12a) are mere functions of \( z^L \). Furthermore, the output delay chain of the polyphase interpolator for summation of the
branch signals is, in compliance with (11a), merely composed of $z^M$-terms. Note that both these features are necessary for the steps to follow.

Replacing $l$ by $m$ and $p$ by $q$ and interchanging $L$ with $M$ in the rearrangement procedure of Step 2, we introduce the corresponding index substitution for Case b:

$$m: = (mL)_M = (mL) \mod M = mL - q_m M \quad (7b)$$

where $m = 0,1,\ldots,M-1$, $q_m \in \mathbb{N}_0$ and correspondingly $q_0 = 0 \leq q_1 \leq \ldots \leq q_m \leq \ldots \leq q_{M-1}$. Instead of (8a) we get the bounds for $q_{M-1}$

$$\frac{(M-1)(L-1)}{M} \leq q_{M-1} \leq \frac{(M-1)L-1}{M} \quad (8b)$$

yielding

$$q_{M-1} = \left\lfloor \frac{(M-1)L-1}{M} \right\rfloor = L - \left\lceil \frac{L+1}{M} \right\rceil \quad (9b)$$

By introducing the index mapping (7b) into (6b), and by setting

$$\mu = q_{M-1} M \quad (10b)$$

under consideration of (9b), we obtain

$$H(z) = z^{-q_{M-1}M} \sum_{m=0}^{M-1} H_{(mL)_M} (z^M) \cdot z^{-(mL-q_m M)} = \sum_{m=0}^{M-1} R_m (z^M) \cdot z^{-mL} \quad (11b)$$

where the quantities

$$R_m (z^M) = H_{(mL-q_m M)} (z^M) \cdot z^{-(q_{M-1}-q_m)M} \quad (12b)$$

represent the $M$ reordered polyphase components of the FSRC output decimator. The decomposed filter $H(z)$ imbedded in-between an input upsampler and an output downsampler is depicted in Fig.2b.

The features of the rearranged polyphase decomposition of Case b correspond to those of (11a) and (12a). Again, the extra delay introduced by the choice of $K = \mu$ in compliance with (9b) and (10b) is minimum. Here, however, the components of (12b) are mere functions of $z^M$, and the tapped input delay chain of the polyphase decimator is, in compliance with (11b), merely composed of $z^L$-terms (cf. Fig.2b).
Step 3: Application of Noble Identities [1]

The structure obtained so far for Case a by decomposition of \( H(z) \) according to (11a) and Fig.2a represents a polyphase input interpolator with subsequent sample rate compression by \( M \), such that all \( L \) branch filters (12a) receive identical input signals upsampled by \( L \) \([1,2]\). Hence, by applying the noble identities [1] to the branch filters and the output delay chain of (11a), the input upsampler by \( L \) and the output downsampler by \( M \) are shifted into the decomposed FSRC implementation until they are directly cascaded at the output of each polyphase branch filter. Reversing the sequential order of the expander-compressor cascade, being admissible since \( L \) and \( M \) are assumed coprime [1], results in the transformed structure as shown in Fig.3a. The associated polyphase decomposition, following from (11a), is given by

\[
H(z) = \sum_{l=0}^{L-1} z^{-LM} E_l(z^L) = \sum_{l=0}^{L-1} z^{-l} E_l(z_1)
\]

(13a)

where the \( z \)-domain variables are related to the operating sampling rates according to (3a,b).

In Case b the decomposed FSRC filter in conjunction with output downsampling by \( M \) according to (11b) and Fig.2b represent a polyphase output decimator such that all \( M \) branch filters operate on a common summing point. Applying the same identity transformations as in
Case a, yields the structure depicted in Fig.3b and the associated polyphase decomposition deduced from (11b):

\[
H(z) = \sum_{m=0}^{M-1} R_m(z^M) \cdot z^{-mL} = \sum_{m=0}^{M-1} R_m(z_o) \cdot z_i^{-m}
\]  

(13b)

**Step 4: Polyphase Decomposition of Branch Filters**

The dashed blocks in Fig.3 represent either decimators for \(M\)-fold sample rate reduction (Case a) or interpolators for \(L\)-fold sample rate increase (Case b). Hence, each of the associated branch filters is subjected to a further polyphase decomposition into \(M\) or \(L\) polyphase components, respectively, with subsequent standard application of the noble identities to polyphase decimators or interpolators [1].

As a result, we obtain for Case a (Fig.3a) \(L\) polyphase decimators with identical compressor-delay chain circuits for blocking their common input signal \(X(z_i)\) into \(L\) identical vector-valued signals of length \(M\), and one output expander-delay chain for interleaving (unblocking) \(L\) branch signals. In Case b (Fig.3b) we have one compressor-delay chain for blocking \(X(z_i)\) into a length-\(M\) signal, and \(M\) polyphase interpolators with identical output expander-delay chains for interleaving \(L\) branch signals each, where all these unblocking circuits feed the same summing point. Finally, the \(L\) identical input blocking circuits or the \(M\) identical output circuits for unblocking, respectively, are replaced by one each by means of elementary signal flow graph identities. Thus, a common block structure results for Case a and Case b, as shown in Fig.4. Nevertheless, for Case a and Case b different MIMO LTI systems \(S(z_s)\) will result since, according to (10a,b), we always have \(\lambda \neq \mu\), to be proved subsequently.

Applying the described algorithmic optimisation to the polyphase representations (13a,b) yields:

\[
H(z) = \sum_{l=0}^{L-1} z_o^{-l} E_l(z_i) = \sum_{l=0}^{L-1} z_o^{-l} \sum_{m=0}^{M-1} E_{lm}(z_s) \cdot z_i^{-m}
\]  

(14a)
\[
H(z) = \sum_{m=0}^{M-1} R_m(z_o) \cdot z_i^{-m} = \sum_{m=0}^{M-1} \sum_{l=0}^{L-1} z_o^{-l} R_{ml}(z_i) \cdot z_i^{-m}
\]  (14b)

Again, these decompositions are related to the actual \( z \)-domain variables or sampling rates, respectively, used for filtering \((z_s, F_s)\), blocking \((z_i, F_i)\) and unblocking \((z_o, F_o)\) operations, as defined by \((3a,b,c)\) and \((4)\).

**Step 5: Commutator Based Minimal Polyphase Structure**

In Fig. 5 the 1-to-\( M \) input blocking and the \( L \)-to-1 unblocking circuits of Fig. 4 are replaced by equivalent commutator models rotating counterclockwise as appropriate [1,2]. Note that the numbering of the commutator positions is related to the type 1 polyphase decomposition of the respective (branch) filters. (If related to the input sequence, the numbering of the input commutator would indicate a type 2 polyphase decomposition of the input signal.) Furthermore, the detailed structures of the \((4)\) resulting from the described twofold polyphase decomposition, are revealed for **Case a** and **Case b**. The corresponding FSRC representation by \((14a,b)\) is still valid.

**B. Discussion**

We restrict our discussion of the above FSRC algorithms to the application of general non-recursive filters because of their widespread use for sample rate alteration. To this end, let

\[
H_c(z) = \sum_{k=0}^{n} h^c(k) z^{-k}
\]  (15)

be a canonical (i.e. minimum delay) causal FIR filter.
Minimal Realisation

By the described twofold polyphase decomposition procedure the $N = n+1$ coefficients of $H_c(z)$ are systematically assigned to the $L \times M$ polyphase branches. Obviously, the resulting multiplication rate is given by

$$F_{\text{ops}} = N \cdot F_s = N \cdot \frac{F_i}{M} = N \cdot \frac{F_o}{L}$$

being likewise minimum for both decomposition strategies. However, the extra delay of $KT$ introduced according to (5) for causality is always different for Case a and Case b. Hence, the minimal polyphase implementation of FRSC is definitely represented by the approach calling for minimum extra delay, defined by

$$K_{\text{min}} = \min \{\lambda, \mu\}$$

with $\lambda, \mu$ according to (10a,b). Correspondingly, as it is generally proved in the appendix, the minimal polyphase representation of FRSC is unambiguously defined by:

$$L < M \iff K_{\text{min}} = \lambda < \mu$$
$$M < L \iff K_{\text{min}} = \mu < \lambda$$

As a result, the minimal implementation of FSRC is always obtained by Case a development (starting with $L$-branch input interpolator decomposition) if $L < M$, and Case b development (starting with $M$-branch output decimator decomposition) if $L > M$, respectively.
Application of Nyquist Filtering

Next, let us assume that $H_c(z)$ of (15) represents a linear-phase $L$th band FIR filter (Nyquist filter) [8,9] of odd length $N$. Properly scaled by the interpolation factor $L$ [1], its impulse response is characterised by the following pre-fixed coefficients:

$$h^c(k) = \begin{cases} 
1, & k = \frac{N}{2} \\
0, & k = \frac{N}{2} \pm \kappa L, \quad \kappa = 1,2,\ldots 
\end{cases} \quad (19)$$

As a result, for some particular branch $l = l$ the polyphase component $E_l(z_i)$ of the first decomposition (cf. Fig.3a) degenerates to a mere delay without any multiplier. Hence, the polyphase decomposition of $E_l(z_i)$, as depicted in Fig.5a, simplifies to just one multiplier-free delay path from the particular position $m = m$ of the input commutator to the position $l = l$ of the output commutator. Note that, due to this singular unweighted input-output commutator connection, every $M$th input sample to the FSRC is directly transferred to its output without being changed.

The same interpolation property is obtained with the structure of Case $b$. Here, however, the coefficients prescribed by (19) are distributed amongst all $M$ branch filters $R_m(z_\circ)$ of the first polyphase decomposition (Fig. 3b), resulting in (cf. Fig.5b):

$$R_m(z_\circ) = \begin{cases} 
1, & m = m \\
0, & \forall m \neq m 
\end{cases} \quad (20)$$

If we used an $M$th rather than an $L$th band filter, with $L$ replaced by $M$ in (19), the above input sample preserving property of the FSRC would be destroyed. This is easily deduced from Fig.5b by.

Transposition and Duality

The transposed structure of a linear time-invariant (LTI) system retains the original transfer function, the number of delays, multipliers and adders, and thus, in particular, the respective original minimality [6,10]. The same holds for linear periodically time-variant (LPTV), i.e. synchronous multirate systems according to Fig.4 with one essential exception: By transposing an LPTV system its original (bifrequency transfer) function is replaced by its dual [10,11] where, for instance, dual pairs are represented by upsampling/downsampling, blocking/unblocking and interpolation/decimation.

Relating this framework to our FSRC, by transposition an $L/M$ interpolator (decimator) is transformed to an $M/L$ decimator (interpolator). Hence, the $M$-input $L$-output LTI system $S(z)$ of Fig.4 is transformed to the corresponding $M\times L$ LTI system with $L$ input and $M$ output ports. In Fig.5 this is reflected by the transposed forms of the branch filters $E_{lm}(z)$ and $R_m(z_\circ)$ in conjunction with reversed signal flows. As a result, all branching nodes of Fig.5 are replaced by summing nodes and vice versa. The blocking and unblocking devices of Fig.4 are subjected to the same rules in conjunction with the formal reversal of the arrows of the blocks for down- and upsampling.

From these considerations we draw the following conclusions:

- Transposing a minimal polyphase realisation of FSRC for sample rate alteration by $L/M$ yields the dual polyphase realisation of FSRC for sample rate conversion by $M/L$. The dual structure is likewise minimal. If the original FSRC is developed along the lines of Case $a$ (Case $b$), then its dual exhibits the structure of Case $b$ (Case $a$), which is in compliance with the minimality condition (18).
• From the above observation readily follows that the alternative implementations of FSRC by \(L/M\), as depicted in Fig.5, are neither transposes nor duals with respect to each other.

• Cascading an FSRC for sample rate alteration by \(L/M\) with its transpose yields a digital system with identical input and output sampling rates. After careful removal of the directly cascaded \(L\)-point output and input commutators in the system centre, such an approach represents a parallel multirate implementation of a monorate LTI system. As a result all operations, except for input and output commutation, are performed at the rate \(F_i/M = F_o/M\). Most importantly, this cascade combination implies the potential of a more general approach to FSRC without any restrictions other than \(L, M \in \mathbb{N}_0 [5,14]\).

**System Control and Latency**

System control is best explained from the block processing representation of FSRC according to Fig.4: A block of \(M\) succeeding samples of the input sequence is concurrently passed to the \(M\) input ports of the MIMO LTI system every \(T_s = MT_i\) seconds at \(kT_s = kMT_i, k = \ldots,-1,0,1,\ldots\ This sampling scheme is reflected by the input vector based on type 2 polyphase decomposition [1]:

\[
x_i(kT_s) = [x(mT_i) \quad x[(m-1)T_i] \quad \ldots \quad x[(m-M+1)T_i]]^T
\]

In compliance with Fig.4, the resulting \(L\) output samples of \(S(z)\) should be available and concurrently be transferred to the unblocking delay chain circuit for interleaving at the same instants \(kT_s\). This is reflected by the output vector applying type 1 polyphase decomposition [1]:

\[
y_o(kT_s) = [y(lT_o) \quad y[(l+1)T_o] \quad \ldots \quad y[(l+L-1)T_o]]^T
\]

So far we have tacitly assumed that downsampling by \(M\) and upsampling by \(L\) is synchronised at \(kT_s = kMT_i = kLT_o, k = \ldots,-1,0,1,\ldots\ (cf. Fig.4). However, for physical realisability that accounts for latency of arithmetic operations [6], in front of each input port of the MIMO system an extra delay element must be inserted. Hence, for a feasible system, (22) is replaced by

\[
y_o[(k-1)T_s] = [y[(l-L)T_o] \quad y[(l-L+1)T_o] \quad \ldots \quad y[(l-1)T_o]]^T
\]

In the implementation of Fig.5 the (un-)blocking circuits of Fig.4 are substituted by equivalent commutator models, where both approaches are required to exhibit identical input-output performance. For this reason, the input commutator has to scan all positions from \(m = M-1\) to \(m = 0\) before calculation of \(y_o[(k-1)T_s]\) can start (1-to-\(M\) blocking operation). Here it is still more obvious that, for physical reasons, each outgoing branch of the input commutator must be provided with a memory for intermediate storage. Compliant with the above discussion of Fig.4, input and output commutator positions \(m = 0\) and \(l = 0\) always coincide (are synchronised), since \(T_s = MT_i = LT_o\) according to (4).

Finally, we want to determine the extra delay introduced by a physically realisable implementation of our block processing approach to FSRC that exceeds the group delay of the FSRC filter \(H_c(z)\). Obviously, this extra delay is composed of two contributions

\[
\Delta \tau = KT + T_s = KT + MT_i = KT + LT_o = (K + LM)T,
\]

where the first term is necessary for system causality, and the second term is due to physical feasibility. In particular, using (10a,b) in conjunction with (1), \(KT\) and, hence, \(\Delta \tau\) is given by:
As a result, the minimal polyphase block processing representation of FSRC according to (18) not only requires the least number of extra memory units, but also gives rise to the minimum overall group delay.

### III. Example and Further Discussion

With $L = 3$, $M = 5$ and

$$H_z(z) = \sum_{k=0}^{n} h^*(k) z^{-k} = \sum_{k=0}^{2n} h_k z^{-k}$$

we have chosen a simple yet general enough FIR-example to gain maximum insight. Due to the above filter order, all entries of the matrix $S(z_s)$ of the $L \times M$ LTI MIMO system of Fig.4 are non-zero, since $N = n+1 = 23 > L \times M = 15$. Hence, each path of both cases of Fig.5 is provided with at least one non-zero coefficient as a result of the twofold polyphase decomposition. We also consider the case of an $L$th band filter, where in compliance with (19):

$$h^*(\frac{\pi}{2}) = h_{11} = 1 \text{ and } h^*(\frac{\pi}{2} \pm \kappa L) = h_{11 \pm \kappa} = 0, \kappa = 1,2,3$$

We start with the development of the minimal implementation defined by Case $a$ according to (18). From (9a), (10a) and (17) follows $p_{L-1} = p_2 = 3$ and $\lambda = K_{\text{min}} = 9$, corresponding to the short extra delay of $9T = 3T_i$. Index mapping (7a) after the first polyphase decomposition is represented by:

$$l \quad 0 \quad 1 \quad 2$$
$$p_L \quad 0 \quad 1 \quad 3$$
$$l^2M \quad 0 \quad 2 \quad 1$$

resulting in the assignment defined by (12a)

$$E_0(z_i) = z_i^3 H_0(z_i) = \sum_{k=0}^{2n} h_{3k+2} z_i^{-(k+3)} = \sum_{k=0}^{S_0} e_{0,k} z_i^{-k}, \quad S_0 = \left\lceil \frac{22}{3} \right\rceil + 3 = 10$$
$$E_1(z_i) = z_i^2 H_2(z_i) = \sum_{k=0}^{2n} h_{3k+2} z_i^{-(k+2)} = \sum_{k=0}^{S_1} e_{1,k} z_i^{-k}, \quad S_1 = \left\lceil \frac{22}{3} \right\rceil + 2 = 9$$
$$E_2(z_i) = H_1(z_i) = \sum_{k=0}^{2n} h_{3k+1} z_i^{-k} = \sum_{k=0}^{S_2} e_{2,k} z_i^{-k}, \quad S_2 = \left\lceil \frac{22}{3} \right\rceil = 7$$

where use is made of (3a) and (6a). Note that some of the leading coefficients of the polyphase components $E_l(z_i)$ for $l < L-1 = 2$ are zero due to the extra delay introduced by (5).

In the second decomposition process defined by (14a) the quantities $E_l(z_i)$ of (29) are additionally decomposed into $M = 5$ polyphase components each. To this end the summation index $k$ used in (29) is split according to

$$k := Mv + m, \quad \text{where } m = 0,1,\ldots,M-1 \text{ and } v = 0,1,\ldots, \left\lceil \frac{k}{M} \right\rceil$$

\[\text{(25)}\]
yielding

\[
E_{lm}(z_s) = \sum_{v=0}^{S_l} e_{l,5v+m} z_{1}^{-(5v+m)} = z_{1}^{-m} \sum_{v=0}^{S_l} e_{l,5v+m} z_{s}^{-v}
\]

(31)

where, in compliance with (29), \( l = 0, 1, 2 \) and \( S_0 = 10, S_1 = 9 \) and \( S_2 = 7 \). The ultimately obtained signal flow graph of the optimum fractional sample rate converter (\( =3/5\)-decimator) is depicted in Fig.6a.

Taking account of the particular coefficients of an \( L \)th band filter according to (27) leads to the deletion of the dashed branches of Fig.6a. Obviously, all inputs to position \( m = 0 \) of the input commutator are unaffectedly transferred to position \( l = 1 \) of the output commutator. For this \( L \)th band case we present the matrix \( S_a(z_s) \) of the \( L \times M \) LTI MIMO subsystem of FSRC (Fig.4):

\[
S_a(z_s) = \begin{bmatrix}
  h_6 z_s^{-1} + h_{21} z_s^{-2} & h_9 z_s^{-1} & h_{12} z_s^{-1} & h_0 + h_{15} z_s^{-1} & h_3 + h_{18} z_s^{-1} \\
  z_s^{-1} & 0 & 0 & 0 & 0 \\
  h_1 + h_{16} z_s^{-1} + h_4 + h_{19} z_s^{-1} & h_7 + h_{22} z_s^{-1} & h_{10} & h_{13}
\end{bmatrix}
\]

(32)

Next, without going too much into detail, we develop the same example along the lines of **Case b**, which will not result in a minimal implementation due to (18). From (9b), (10b) and
(17) follows $q_{M+1} = q_4 = 2$ and $\mu = \mu_{\text{min}} = 10 > K_{\text{min}} = 9$, corresponding to an extra delay of $10T = 2T_o$. Performing index mapping and twofold polyphase decomposition according to (7b), (12b) and (14b), respectively, by replacing $p,e,E$ with $q,r,R$ and interchanging $L,L,z_1$ with $m,M,z_o$ in (28) - (31), yields the alternate signal flow graph of the 3/5-decimator as depicted in Fig.6b. Again, the $L$th band filter case is obtained by deleting the dashed branches. Here, in contrast to Case a development, the direct input-output connection starts from position $m = 2$ of the input commutator and ends at position $l = 0$ of the output commutator. As expected, the associated matrix $S_o(z_s)$ of the $L \times M$ LTI MIMO subsystem of FSRC

$$
S_b(z_s) = \begin{bmatrix}
0 & 0 & z_s^{-1} \\
h_{10}z_s^{-1} & h_{13}z_s^{-1} & h_1 + h_{16}z_s^{-1} & h_4 + h_{19}z_s^{-1} & h_7 + h_{22}z_s^{-1} \\
h_0 + h_{15}z_s^{-1} & h_3 + h_{18}z_s^{-1} & h_6 + h_{21}z_s^{-1} & h_9 & h_{12}
\end{bmatrix}
$$

(33)

is totally different from $S_o(z_s)$ of Case a (32), since $\lambda \neq \mu$. However, $S_o(z_s)$ is readily obtained by a Case a development starting with $K = \lambda + 1 = \mu = 10$, where the FSRC filter (5) is replaced with:

$$
H'(z) = z^{-\lambda - 1}H_o(z) = z^{-\lambda} \frac{H(z)}{z} = z^{-\lambda}H'_e(z)
$$

(34)

Note that Fig.6b is easily redrawn in structural terms of Case a. Generalizing this modification of (5), it is obvious that Case a and Case b development is likewise possible for any number of extra delays, provided that $K \geq \lambda$ or $K \geq \mu$, respectively, where $\lambda$ and $\mu$ are defined by (10a,b).

Structural development with arbitrary supplementary delay regarding the above constraints is, however, more conveniently achieved by manipulation of the subsystem matrix $S(z_s)$, as it will be shown subsequently in conjunction with Case a (32). To start this investigation, assume a delay by $T_i$ in front of the 1-to-$M$ blocking circuit of FSRC (Fig.4). In the commutator model based approach (Fig.5) such a delay is immediately realised by redefining a complete cycle of the input commutator according to: $m \in \{M-2, M-3, ..., 1, 0, M-1\}$ with starting at position $m = M-2$ and ending at $m = M-1$. The resulting somewhat peculiar synchronisation scheme (input-output clock coincidence at positions $m = 0/l = 0$ is destroyed) can be circumvented if, instead, the columns of $S(z_s)$ are circularly shifted to the right by one position, while the originally rightmost column becomes the leftmost one. This column shift (equivalently replacing commutator position shift) correctly accounts for the delay by $T_i$ (z-domain multiplication by $z_i^{-1}$) except for the new leftmost column. The latter shift gives rise to a time advance corresponding to $z_i^{M-1}$, which has to be compensated by a multiplication in z-domain by $z_i^{-M} = z_s^{-1}$. As an example, a twofold circular column shift of $S_a(z_s)$ given by (32) to the right yields:

$$
S_a^{**}(z_s) = \begin{bmatrix}
h_0z_s^{-1} + h_{15}z_s^{-2} & h_3z_s^{-1} + h_{18}z_s^{-2} & h_6z_s^{-1} + h_{21}z_s^{-2} & h_9z_s^{-1} & h_{12}z_s^{-1} \\
h_0z_s^{-1} & h_{13}z_s^{-1} & h_1 + h_{16}z_s^{-1} & h_4 + h_{19}z_s^{-1} & h_7 + h_{22}z_s^{-1}
\end{bmatrix}
\Rightarrow S_a(z_s)z_i^{-2}
$$

(35)

where a supplementary delay by $2T_i = 6T$ is introduced. Applying (1), (5) and (10a,b), respectively, the overall extra delay is expressed in more general terms:

$$
KT = (\lambda + 6)T = (3L + 2L)T = LMT = (2M + M)T = (\mu + M)T = 15T = MT_i = LT_o
$$

(36)
Note that, as a result of the above matrix manipulation, all entries of the top row of (35) are provided with extra storage (representing one leading zero-coefficient).

Similarly, time advance or reduction of excess input delay, respectively, is obtained by circular shift of the MIMO subsystem matrix columns to the left and by multiplying the transfer functions of the skipping columns by \( z_s \). Obviously, any column leftshift of (32) would result in a non-causal system, since (32) represents the minimal implementation of FSRC with all entries of the bottom row free of extra storage.

Time delay or time advance by integer multiples of \( T_o \), introduced behind the \( L \)-to-1 un-blocking circuit of FSRC, can likewise be achieved by manipulation of the subsystem matrix \( S(z_s) \). Time advance by \( T_o \), for instance, corresponds to a redefinition of a full cycle of the output commutator according to \( l \in \{1, 2, ..., L-1, 0\} \) or, equivalently, by upshifting the matrix rows by one position and by multiplying the transfer functions of the skipping rows by \( z_s \). In continuation of the above example the rows of \( S_a''(z_s) \), as given by (35) after twofold column shift, are shifted upwards by one position

\[
\begin{bmatrix}
0 & 0 & z_s^{-1} \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
h_{10}z^{-1}_s & h_{13}z^{-1}_s & h_{1} + h_{16}z^{-1}_s & h_{4} + h_{19}z^{-1}_s & h_{7} + h_{22}z^{-1}_s \\
h_{0} + h_{13}z^{-1}_s & h_{3} + h_{18}z^{-1}_s & h_{6} + h_{21}z^{-1}_s & h_{9} & h_{12}
\end{bmatrix}
\Leftrightarrow S_a'(z_s) \frac{z_s}{z_i} \tag{37}
\]

corresponding to a time advance by \( T_o = 5T \), which partly compensates for the excess delay introduced by column shifting according to (35) and (36). Hence, the reduced overall extra delay is now given by

\[
KT = LMT - T_o = (L-1)T_o = (L-1)MT = 2MT = \mu T = 10T \tag{38}
\]

From (38) we conclude that, by shifting the columns and rows of the MIMO subsystem matrix \( S_a(z_s) \) appropriately, the optimum Case b transfer matrix \( S_b(z_s) \) can be derived, which is confirmed by comparison of (37) with (33). Note that, since the matrix examples (33) and (37) are all based on the assumption of an \( L \)th band filter design as given by (27), by an additional circular upshift of the rows of (37) a causal implementation with an extra delay below the limits of (36) and (38) is feasible: \( KT = 5T \). Due to (18), this is not possible for general filter designs.

Next, we want to take a closer look at the hardware expenditure of the two implementations of the 3/5-decimators according to Figs.6a,b. Both approaches are relatively optimal, since \( \lambda \) and \( \mu \) were assigned their minimum possible values according to (10a,b). Since \( \lambda < \mu \), Case a development (Fig.6a) represents the minimal approach. Hence, the overall group delay difference of both implementations is always given by (\( \mu-\lambda \) ) \( T \), which is independent of all structural manipulations to follow. Obviously, by direct comparison of the realisations of Figs.6a,b, the hardware expenditure of both structures is identical except for the extra delay elements originally introduced for causality reasons in (5):

\[
\begin{align*}
\text{Case a:} & \quad 5 \text{ extra delays in branches } l = 0 / m = 0,1,2; \quad l = 1 / m = 0,1 \\
\text{Case b:} & \quad 6 \text{ extra delays in branches } l = 0 / m = 0,1,2,3; \quad l = 1 / m = 0,1
\end{align*}
\]

It is noteworthy to recognise that this slight hardware advantage of Case a is transferred to Case b if the system applies an \( L \)th band filter design, where all dashed branches of Figs.6a,b are removed:
Casea: 4 extra delays in branches \( l = 0 / m = 0,1,2; \ l = 1 / m = 0 \)

Caseb: 3 extra delays in branches \( l = 0 / m = 2; \ l = 1 / m = 0,1 \)

These extra delay elements can be found in the transfer matrix \( S_a(z_s) \) according to (32) in the first row (entry 1,2,3) and in the second row (1st entry), and in \( S_b(z_s) \) according to (33) in the first two rows (3rd entry and entry 1,2, respectively). Common extra delays of a row (column) can be removed and replaced by just one in that row (column), as shown for Case a (32):

\[
S_a(z_s) = \begin{bmatrix}
    z_s^{-1} (h_6 + h_{21} z_s^{-1}) & h_9 & 0 & 0 & 0 & 0 & 0 \\
    z_s^{-1} (1) & h_{12} & h_0 + h_{15} z_s^{-1} & h_3 + h_{18} z_s^{-1} & 0 & 0 & 0 \\
    h_1 + h_{16} z_s^{-1} & h_4 + h_{19} z_s^{-1} & h_7 + h_{22} z_s^{-1} & h_{10} & h_{13} & 0 & 0 \\
\end{bmatrix}
\]  

As a result, in contrast to \( K \) according to (10a,b), only \( L-1 \) (by rowwise removal) or \( M-1 \) (by columnwise removal) extra delay elements are likewise required for Case a and Case b implementations, respectively, since the bottom row of the transfer matrix is always completely free of extra delays (cf. Fig.2).

Dramatic reduction of hardware for signal storage, however, is achieved by merging all \( L \) branch filter state variable memory sets of each of the \( M \) positions of the input commutator into one delay set only, as shown by the rearranged Case a structure depicted in Fig.7.
The application of this procedure to Case \( b \) (Fig.6b) leads to identical hardware requirements, whereas assignment of coefficients to branches is totally different. As a general result, the number of delay elements needed (excluding those \( M \) necessary for physical feasibility) for this rearranged structure is upper bounded by

\[
\begin{align*}
n_D & \leq M \left\lfloor \frac{N + K}{L \cdot M} \right\rfloor - 1
\end{align*}
\]

where the quantity in brackets represents the number of delays of a single branch filter, \( N \) the length of the FIR filter, and \( K \) is given by (10a,b). Obviously, the more \( N + K \) exceeds \( L \times M \), the higher the savings of storage. In our examples (Figs.6a,b) the original number of 13 (Case \( b \): 14) delays is diminished to 6 (\( n_D \leq 10 \)). Finally, it should be noticed that the structural approach of Fig.7 is particularly well suited for coefficient hardwiring in conjunction with the ternary Canonical Signed Digit (CSD) code for coefficient representation [15].

Fig. 7: Input-oriented memory merging, equivalent to Fig.6a
Alternatively, it is also possible to merge all $M$ branch filter state variable memory sets of each of the $L$ positions of the output commutator into one delay set only. To this end, in a preparatory step all $L \times M$ branch filters of Figs.6a,b must be transposed individually before restructuring. The ultimate result of this rearrangement procedure, applied to the Case a example, is shown in Fig.8. Hence, the upper bound (40) is replaced by:

$$n_D \leq L \left[ \frac{N + K}{L \cdot M} - 1 \right]$$  \hspace{1cm} (41)
Thus, in our example we only need four state variable delays ($n_D \leq 6$). The additional storage necessary for physical feasibility is, for completeness and awareness, included in Fig.8. Note that, for our 3/5-decimator example, Fig.8 represents the absolutely minimal polyphase implementation of FSRC, which is characterised by the least possible number of delay elements and the lowest overall group delay in conjunction with block processing.

As a further result of the above described restructuring and the associated discussion, it is obvious that the issue of a minimum number of delay units in conjunction with time-
multiplexing of subsystems of the FSRC MIMO system stressed in [17] has completely been overcome.

Finally, we make some additional remarks about transposition of the block processing approach to FSRC. Structural transposition is readily obtained by reversing all arrows of Figs.6-8 (signal flow and sense of commutator rotation), which implies matrix transposition of the \( L \times M \) LTI MIMO subsystem \( S_{L \times M}(z) \) of FSRC to \( S_{M \times L}(z) \). As described in section II.B, the 3/5-decimator is transformed to its dual 5/3-interpolator. As expected [10], the original structural optimality (minimum computational burden) is retained, as it is immediately seen by transposing Figs.7 and 8. Again, the excess group delay of the transposed system is given by (24).

In all examples of Figs.6-8 the dashed branches are discarded, if the FSRC filter (5) represents an \( L \)th band filter according to (27). As a consequence of this particular design, every \( M \)th input sample to the FSRC system is unaffectedly transferred to the output. This sample preserving property is destroyed by transposition, which is readily confirmed by inspection of the examples Figs.6-8. This is also reflected by the fact that the particular output related row vector of \( S(z) \) which contains only one non-zero element, for instance the 2\(^{nd} \) row of (32), is transposed to an input related column vector. Hence, if the sample preserving property is desired for the dual FSRC system as well, \( H(z) \) has to be designed anew as an \( M \)th band filter. A new filter design for the transposed structure might, in general, also be necessary, if the filter requirements of the dual FSRC pair turned out to be different. In any case, the FIR filter coefficients must be rescaled by \( M/L \) after transposition.

IV. Conclusion

In this paper a novel systematic and rigorous 5-step derivation of fractional sample rate conversion (FSRC; Fig.1) of synchronous sampling frequencies by a fixed ratio \( L/M \) is given, where \( L \) and \( M \) are constrained to be relatively prime integers. The correctness of the algorithms has been verified by computer simulation using MATLAB. The main features of the described approach to FSRC, elaborated by general investigation (section II) and detailed discussion of various examples (section III), are listed subsequently:

- The derivation applies twofold polyphase decomposition of a given FSRC filter (Fig.1) in the frequency domain, allowing for two different implementations (options) in dependence of the sequential order of the polyphase decompositions.
- Our derivation and implementation of FSRC is based on block processing: Non-overlapping blocks of \( M \) succeeding input samples each are formed and subjected to concurrent (parallel) processing.
- Additional memory is introduced to guarantee causality of FSRC in conjunction with block processing, giving rise to a somewhat increased overall system group delay. Conditions for minimum extra delay are given and proved for both options, where minimum extra delay of both options is always different.
- Interfacing sequential transmission and parallel processing only requires a single 1-to-\( M \) input blocking circuit and a single output \( L \)-to-1 unblocking (interleaving) circuit, where both of which can be replaced by equivalent commutator models.
- Due to block processing all operations (multiplication, addition and storage of state variables) are performed at subnyquist rate ranging below input and output rates (4).
- Both of the above options of FSRC can, although different, be implemented with exactly the same hardware expenditure where, as a result of structural optimisation, the number of delay elements needed is much less than the order of the FSRC filter. Hence, depending on speed requirements, these structures lend themselves to time-multiplexing of subunits (multipliers) of the LTI MIMO subsystem. Alternatively, most efficient high-speed implementations applying CSD code representation of coefficients are feasible.
A versatile matrix representation of the $L \times M$ LTI MIMO subsystem $S(z)$ of FSRC is given. By combined row and column shifting, both of the above options can be derived from each other. Moreover, any desired group delay in excess of the minimum value dictated by causality (feasibility) can flexibly be achieved by simple manipulation of this matrix representation. Nevertheless the hardware expenditure is, within wide limits, independent of the prescribed group delay.

The maximum possible number of input samples to the system is retained in its output sequence (sample preserving property), if FSRC implementation applies $L$th band (Nyquist) FIR filter designs [8].

Transposing a minimal system for FSRC by $L/M$ results in the corresponding dual system for FSRC by $M/L$ being likewise minimal [10]. In addition to the results reported, we have also discussed issues on system control. Moreover, the system excess group delay was determined under consideration of latency due to physical realisability.

Concurrently, as will be published elsewhere [14], a derivation of a more general block processing approach to FSRC has been developed that overcomes the constraint of $L$, $M$ to be relatively prime. Moreover, the extra delay of (5) for system causality is avoided by an approach to FSRC applying sample-by-sample rather than block processing, which has been derived by twofold time-domain polyphase decomposition of the system-theoretic approach according to Fig.1 [16].

**Acknowledgement**

This work was supported by Deutsche Forschungsgemeinschaft under contract GO 849/1-1.

**Appendix: Prove of minimum extra delay conditions.**

Introducing (9a,b) and (10a,b) into the minimum delay inequality $\lambda < \mu$ of Case a yields:

$$\lambda = ML - \left[ \frac{M+1}{L} \right] L < LM - \left[ \frac{L+1}{M} \right] M = \mu$$

or equivalently

$$\left[ \frac{M+1}{L} \right] L > \left[ \frac{L+1}{M} \right] M$$

(A1)

First assume $1 \leq L < M$. Hence $\forall L < M$, $M = L + \Lambda$, where $\Lambda \in \mathbb{N}$, and $\left[ \frac{L+1}{\Lambda} \right] = 1$. Introducing these relations into (A1) it readily follows:

$$1 + \frac{\Lambda+1}{L} L = L + \left[ \frac{\Lambda+1}{L} \right] L > L + \Lambda = M$$

(A2)

verifying $\lambda < \mu$ for $L < M$. In order to prove that this sufficient condition is also necessary, we next assume $1 \leq M < L$. Hence $\forall M < L$, $L = M + \Xi$, where $\Xi \in \mathbb{N}$, and $\left[ \frac{M+1}{\Xi} \right] = 1$. By inserting these relationships again into (A1) we get:

$$\left[ \frac{L+1}{M} \right] M = \left[ 1 + \frac{\Xi+1}{M} \right] M = M + \left[ \frac{\Xi+1}{M} \right] M > M + \Xi = L$$

(A3)
being in contradiction to (A1) and, thus, falsifying \( \lambda < \mu \) for \( L > M \). Obviously, \( \lambda = \mu \) is only possible for \( L = M \) being excluded in our derivation procedure.

The necessary and sufficient condition \( M < L \) that meets the minimum delay inequality \( \mu < \lambda \) of Case b is likewise proved by interchanging both \( \lambda \) with \( \mu \) and \( L \) with \( M \) in the above relationships.

As a result, for any \( L < M \) (\( L > M \)) the extra delay is definitely minimum for Case a (Case b).

REFERENCES