Loop Group Factorization of Biorthogonal Wavelet Bases

Silviu Borac
Eigenval Research Inc., Toronto*

Ruedi Seiler
Institute for Advanced Study, Princeton, New Jersey †

November 4, 1997, Revised Version

Abstract

We present a factorization theorem for Laurent-polynomial loops in the matrix group $GL(2, \mathcal{F})$ over a field $\mathcal{F}$ which is part of the complex numbers. The theorem leads to the construction of an algorithm for the factorization of pairs of biorthogonal filters with finite impulse response. An immediate application is the reduction of the complexity of the corresponding wavelet transform.

1 Introduction

Biorthogonal wavelets give rise to perfect reconstruction filters which are useful for audio and image data compression. We propose a method for factorization of the corresponding polyphase matrix in factors of low computational complexity.

*e-mail: silviu@eigenval.com

†On leave from the Technische Universität Berlin, Department of Mathematics, e-mail: seiler@math.tu-berlin.de

The factorization of filters, as a method for diminishing the computational complexity, has been discussed before by several authors, in particular by Nguyen and Vaidyanathan [7], Vetterli [12] and more recently by Daubechies and Sweldens [2].

The factorization of biorthogonal wavelet filters can be interpreted as a factorization of elements in the loop group $\mathcal{L}GL(2, \mathcal{F})$, where $\mathcal{F}$ denotes an arbitrary subfield of the complex numbers. Loop groups are interesting examples of infinite dimensional groups and have arisen in physics in the context of two-dimensional integrable models and string theory [4] [10]. Factorization in loop groups is an important standard tool and the one presented here is related to Birkhoff's factorization. Loop group factorizations have recently found a particularly nice application in the theory of two-dimensional surfaces [3].

The factorization result presented here is not new. It turns out that it is a special case of one of the most prominent results of commutative algebra which states that projective modules over polynomial rings are free. Logar and Sturmfels [6] and Park and Woodburn [8] have given algorithmic proofs for the special case of the ring of polynomials over a field. The importance of this theorem for signal analysis has been discussed in Park [9]. The approaches of Tolhuizen, Hollmann and Kalker [11] and Daubechies and Sweldens [2] are based on the Euclidean algorithm for Laurent polynomials in one variable. Holschneider and Pinkall [5] presented a factorization algorithm based on loop groups for the orthogonal case.

The main result of this paper, is a simple algorithm for the decomposition of biorthogonal filters with finite impulse response into elementary filters, stated as Theorem 3.1. The algorithm is designed for Laurent-polynomial loops over an arbitrary field $\mathcal{F}$ in $GL(2, \mathcal{F})$. It is applicable also to rings of coefficients that are not fields, like the rings of integers and of dyadic rationals in certain cases (see example 4.4). We have not yet been able to characterize the classes of filter pairs that can be factorized in these non-field situations.

The result is based on the observation, that biorthogonal filters with finite impulse response are in one to one correspondence with polynomial loops in $GL(2, \mathcal{F})$. The polynomial loop group $\mathcal{L}_{pol}GL(2, \mathcal{F})$ is graded. For $\mathcal{F} = \mathbb{C}$ the grading is according to the winding number of the determinant map. The shift acts as a ladder operator.

Factorization gives an exhaustive parametrisation of all biorthogonal FIR filters. An algebraic characterisation of filter pairs leading to reagular wavelets is, however, still missing.

The article is structured as follows:
After an introductory first section, the second one is devoted to the connection of biorthogonal filters and loops in $GL(2, \mathcal{F})$. In the third section we explain our method of factorizing loops and in the last one we give an estimate on computational complexity, present some examples and give the pseudo code for the factorization algorithm.

2 From Biorthogonal Filters to Loops

A multiresolution analysis of $L^2(\mathbb{R})$ [1] consists of a a sequence of closed spaces V_j satisfying:

$$\begin{aligned} \cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \\ \overline{\bigcup_{j\in\mathbb{Z}} V_j} = L^2(\mathbb{R}) \\ \bigcap_{j\in\mathbb{Z}} V_j = \{0\} \end{aligned} \tag{2.1}$$

For every j we have a decomposition of V_j into an average space and a detail space:

$$V_j = V_{j-1} \oplus W_{j-1}, \quad j \in \mathbb{Z} \tag{2.2}$$

The multiresolution analysis is built using two basic functions: a scaling function ϕ and a wavelet ψ. The integer translates of the 2^j-dilation of ϕ form an L^2-basis [1] for $V_j = \langle\langle \phi\,(2^j x - m)\,,\ m \in \mathbb{Z}\rangle\rangle$. Similarly, we have $W_j = \langle\langle \psi\,(2^j x - m)\,,\ m \in \mathbb{Z}\rangle\rangle$. Furthermore $L^2(\mathbb{R}) = \langle\langle \psi\,(2^j x - m)\,,\ m \in \mathbb{Z}, j \in \mathbb{Z}\rangle\rangle$. Due to eq.(2.1) the functions ψ and ϕ satisfy two-scale refinement relations

[1] Here, an L^2-basis is what is sometimes called an unconditional L^2-basis.

of the form:

$$V_{-1} \ni \phi\left(\frac{x}{2}\right) = \sqrt{2}\sum h_k \phi\,(x-k)$$
$$W_{-1} \ni \psi\left(\frac{x}{2}\right) = \sqrt{2}\sum g_k \phi\,(x-k) \tag{2.3}$$

The dual basis is given in terms of the so called dual scaling function $\tilde{\phi}$ and dual wavelet $\tilde{\psi}$. They generate another multiresolution analysis of $L^2(\mathbb{R})$:

$$\cdots \subset \widetilde{V}_{-2} \subset \widetilde{V}_{-1} \subset \widetilde{V}_0 \subset \widetilde{V}_1 \subset \widetilde{V}_2 \subset \cdots$$
$$\overline{\bigcup_{j\in\mathbb{Z}} \widetilde{V}_j} = L^2\,(\mathbb{R}) \qquad \bigcap_{j\in\mathbb{Z}} \widetilde{V}_j = \{0\}$$
$$\widetilde{V}_j = \widetilde{V}_{j-1} \oplus \widetilde{W}_{j-1} \tag{2.4}$$

The functions $\phi\,,\ \psi\,,\ \tilde{\phi}\,,\ \tilde{\psi}$ satisfy the following orthogonality relations in $L^2(\mathbb{R})$:

$$\langle\tilde{\phi}\,,\ \psi(\cdot - n)\rangle = \langle\tilde{\psi}\,,\ \phi(\cdot - n)\rangle = 0 \quad n \in \mathbb{Z}$$
$$\langle\tilde{\phi}\,,\ \phi(\cdot - n)\rangle = \langle\tilde{\psi}\,,\ \psi(\cdot - n)\rangle = \delta_{n,0} \tag{2.5}$$

The dual scaling function and wavelet satisfy refinement relations similar to (2.3) with filter coefficients $\tilde{h}_k$ and $\tilde{g}_k$, respectively.

A function $f \in V_0 \subset L^2\,(\mathbb{R})$ can be decomposed in the basis $\{\phi(\cdot - n)\,,\ n \in \mathbb{Z}\}$ as follows:

$$f(x) = \sum_{j\in\mathbb{Z}} f_n\,\phi(x-n) \quad \text{with} \quad f_n = \langle f(x)\,,\ \tilde{\phi}(x-n)\rangle \tag{2.6}$$

where $\langle \quad , \quad \rangle$ denotes the scalar product in L^2. The sequence of coefficients $\{f_n\}$ is in $l^2\,(\mathbb{Z})$. To this sequence corresponds, via the discrete Fourier transform, a square integrable function on the circle $\mathbb{T}$:

$$f(z) = \sum_{n\in\mathbb{Z}} f_n z^{-n}\,, \quad z \in \mathbb{T} = \{z \in \mathcal{F}\,,\ \mid z \mid = 1\} \tag{2.7}$$

We have the sequence of bijections:

$$\begin{array}{rlll} L^2\,(\mathbb{R}) \supset V_0 = \langle\langle\, \phi(\cdot - n) \ \mid n \in \mathbb{Z}\,\rangle\rangle \longrightarrow & l_2(\mathbb{Z}) \longrightarrow & L^2(\mathbb{T}) \\ f(x) = \sum f_n \phi(x-n) \longmapsto & \{f_n\}_{n\in\mathbb{Z}} \longmapsto & f(z) = \sum f_n z^{-n} \end{array}$$

To the filter sequences $h_k,\ g_k,\ \tilde{h}_k,\ \tilde{g}_k$ correspond the functions $h(z),\ g(z), \tilde{h}(z),\ \tilde{g}(z)$.

There is another representation, called the polyphase representation, that is very convenient for expressing wavelet transforms because it diagonalizes the double shift $\{f_n\} \longmapsto \{f_{n-2}\}$ (see below). It is based on the generalized Fourier transformation:

$$\begin{array}{rcc} l_2(\mathbb{Z}) \longrightarrow & & L^2(\mathbb{T}, \mathcal{F}^2) \\ \{f_n\}_{n\in\mathbb{Z}} \longmapsto & & \mathbf{f}(z) = \sum_n z^{-n}\mathbf{f_n} = \sum_n z^{-n}\binom{f_{2n}}{f_{2n+1}} \end{array} \tag{2.8}$$

where $L^2(\mathbb{T}, \mathcal{F}^2)$ is the set of column vectors with two components that are complex functions on the circle, of finite L^2 norm with respect to the product $\langle f, g\rangle = \sum_n \mathbf{f}^*_n \mathbf{g}_n$, where $\mathbf{f}^*_n = \left(\overline{f_{2n}}, \overline{f_{2n+1}}\right)$.

The polyphase representation $\mathbf{f}(z)$ of the sequence $\{f_n\}$ is related to the shift operator S, $Sf_n := f_{n-1}$, through the formula

$$f(z) = \sum_n z^{-n}(S^{-2n}f)_m, \qquad (m = 0, 1) \tag{2.9}$$

$$= \sum_n z^{-n} f_{2n+m} \tag{2.10}$$

We call (as in [2]) the first component of $\mathbf{f}(z) \in \mathcal{F}$ the "even" part and the second component the "odd" part: $\mathbf{f}(z) = \begin{pmatrix} f_e(z) \\ f_o(z)\end{pmatrix}$.

The relation between the $L^2(\mathbb{T})$ representation and the $L^2(\mathbb{T}, \mathcal{F}^2)$ is given by:

$$f(z) = \left(1\, ,\ z^{-1}\right)\mathbf{f}(z^2) \tag{2.11}$$

and

$$\mathbf{f}\left(z^2\right) = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ z & -z\end{pmatrix}\begin{pmatrix} f(z) \\ f(-z)\end{pmatrix} \tag{2.12}$$

In $L^2(\mathbb{T}^1, \mathcal{F}^2)$ the shift operator becomes the following 2×2 matrix:

$$(S\mathbf{f}) = \begin{pmatrix} 0 & z^{-1} \\ 1 & 0\end{pmatrix}\mathbf{f}(z) \tag{2.13}$$

The square of the shift operator is diagonal:

$$\left(S^2\mathbf{f}\right)(z) = \begin{pmatrix} z^{-1} & 0 \\ 0 & z^{-1}\end{pmatrix}\mathbf{f}(z) \tag{2.14}$$

The shift operator is, up to a constant permutation matrix Π, the same as the even and odd translation operator T_e, T_o defined by:

$$T_e = \begin{pmatrix} z^{-1} & 0 \\ 0 & 1 \end{pmatrix}, T_o = \begin{pmatrix} 1 & 0 \\ 0 & z^{-1} \end{pmatrix}, \Pi = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{2.15}$$

$$T_e = S\Pi, \quad T_o = \Pi S \tag{2.16}$$

The forward wavelet transform uses $\tilde{h}$ and $\tilde{g}$ as analysis filters while the inverse wavelet transform uses h and g as synthesis filters. [1]

The conditions for perfect reconstruction are:

$$h(z)\overline{\tilde{h}(z)} + g(z)\overline{\tilde{g}(z)} = 2 \qquad h(z)\overline{\tilde{h}(z)} + g(z)\overline{\tilde{g}(z)} = 0 \tag{2.17}$$

They can be expressed in terms of the polyphase matrices:

$$P(z) = \begin{pmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{pmatrix}, \quad \tilde{P}(z) = \begin{pmatrix} \tilde{h}_e(z) & \tilde{g}_e(z) \\ \tilde{h}_o(z) & \tilde{g}_o(z) \end{pmatrix} \tag{2.18}$$

Perfect reconstruction is achieved when:

$$P(z)\tilde{P}(z)^* = 1 \tag{2.19}$$

The equation above expresses $\tilde{P}$ in terms of P and vice versa. Each polyphase matrix is a loop in the group of invertible two by two matrices $GL(2, \mathcal{F})$. In the case of an orthogonal wavelet, $P(z)$ is equal to $\tilde{P}(z)$. The corresponding loop is in the group of unitary matrices $U(2)$. For the special case of filters with real coefficients and z of modulus one, equation (2.19) becomes:

$$P(z)\tilde{P}^t(z^{-1}) = 1 \tag{2.20}$$

In this article we shall consider only the case of finite impulse filters (FIR). In this case, the entries in both matrices $P(z)$ and $\tilde{P}(z)$ are Laurent polynomials. This implies that det $P(z)$ is also a Laurent polynomial and, by the perfect reconstruction condition (2.19), it is in fact a monomial:

$$\det P(z) = \lambda z^n \tag{2.21}$$

The problem of finding biorthogonal filters with finite impulse response is thus equivalent to finding polynomial loops in $GL(2, \mathcal{F})$. Hence classification of polynomial loops in $GL(2, \mathcal{F})$ is tantamount to classification of finite impulse filters. This is the subject of the following section.

3 Factorization of Loops

A loop group $\mathcal{L}G$ is the group of maps form the circle $\mathbb{T}$ into a group G. The group composition law of $\mathcal{L}G$ is given by the pointwise composition in G [10]. If G is a matrix group, $\mathcal{L}G$ contains the subgroup of polynomial loops $\mathcal{L}_{pol}G$. P belongs to $\mathcal{L}_{pol}G$, if P and its inverse in $\mathcal{L}G$ are polynomials in z.

As mentioned in the previous section, each FIR biorthogonal filter can be identified with a polynomial loop in $GL(2, \mathcal{F})$ For several reasons it is appropriate to consider only loops $P(z)$ which are normalized by $|\det P(z)| = 1, \quad z \in \mathbb{T}$. First, a loop which corresponds to a wavelet, has to satisfy a minimal regularity condition[1]. In terms of $P(z)$ this is

$$P(z=1) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{3.1}$$

Hence, for the class of loops arising from wavelets, $\det P(z=1) = 1$. Second, in the orthogonal case, this condition is automatic. Third, and most importantly, the shift operators S and T_e, T_o belong to this class. Fourth, each finite impulse filter is of this type up to a non vanishing factor $\lambda \in \mathcal{F}$. For the purpose of classification of filters, it suffices to consider the following restricted class of loops:

Definition 3.1. *Let X be the group of Laurent polynomial loops $P : \mathbb{T} \to GL(2, \mathcal{F})$ such that the determinant is a circle map det $P : \mathbb{T} \to \mathbb{T}$. We refer to elements in X as polynomial loops.*

X is clearly a subgroup of $\mathcal{L}GL(2, \mathcal{F})$ and its elements are in one to one correspondence with normalized polyphase matrices of biorthogonal filters with finite impulse response. Due to the following proposition it is graded:

Proposition 3.1. *Let $P \in X$. Then there exist $\lambda \in \mathbb{T}$ and $n \in \mathbb{Z}$ such that det $P(z) = \lambda z^{-n}$.*

Proof. Since $P(z)$ is a Laurent polynomial loop we have det $P(z) = \sum_{n=n_s}^{n_e} a_n z^{-n}$. The leading term of the Laurent polynomial $|$ det $P(z) |^2 = 1$ is $\sum a_{n_s} \overline{a_{n_e}} z^{-n_s+n_e}$ which implies $n_s = n_e$ and $| a_{n_s} | = 1$. $\square$

We get the first structural result about X:

Proposition 3.2. *The loop group X is graded. The grading is compatible with the multiplication:*

$$\begin{array}{llll} X = & \bigcup_n X_n, \quad X_n = & \{P \in X \mid det\, P(z) = \lambda z^{-n}, \lambda \in T\} \\ m: & X_{n_1} \times X_{n_2} & \longrightarrow X_{n_1+n_2} \end{array}$$

The operator S is a ladder operator for the grading: $S: X_n \longrightarrow X_{n+1}$, $S^{-1}: X_{n+1} \longrightarrow X_n$.

As a consequence, for any loop $A \in X$, there is an $n \in \mathbb{Z}$ such that $S^n A$ is a loop with a determinant of modulus one.

It turns out that every polynomial loop is a product of elementary loops of three different types which we call upper triangular, lower triangular and diagonal loops.

Definition 3.2. *Let $k, m, n \in \mathbb{Z}$ and $a \in \mathbb{R}$. Then,*

$$U_+(k,a) := T_e^k \begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} T_e^{-k} = \begin{pmatrix} 1 & z^{-k}a \\ 0 & 1 \end{pmatrix}$$

and

$$U_-(k,a) := T_o^k \begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix} T_o^{-k} = \begin{pmatrix} 1 & 0 \\ z^k a & 1 \end{pmatrix}$$

are special loops in $SL(2, \mathcal{F})$ referred to as upper and lower triangular loops, respectively. They are both unitarily conjugated to the corresponding elementary triangular matrices. Furthermore

$$D(m,n) = \begin{pmatrix} z^m & 0 \\ 0 & z^n \end{pmatrix}$$

is an elementary diagonal loop. It is an element of X_{m+n}.

The following factorization of polynomial loops holds [2]

Theorem 3.1. *Consider a polynomial loop $P \in X$. There exist finitely many upper and lower triangular loops and an elementary diagonal loop such that:*

$P = U_{\sigma(1)} \cdots U_{\sigma(N)} D.$

$\sigma(1) \ldots \sigma(N)$ denotes an appropriate choice of signs.

Before we prove the theorem let us add some remarks and a corollary:

Remark 3.1. The above theorem implies the decomposition of loops using lifting steps in the sense of Daubechies and Sweldens [2], since:

$$U_+(k_1,a_1)\cdots U_+(k_s,a_s) = \begin{pmatrix} 1 & \sum_1^s a_i z^{-k_i} \\ 0 & 1 \end{pmatrix} \tag{3.2}$$

$$U_-(k_1,a_1)\cdots U_-(k_s,a_s) = \begin{pmatrix} 1 & 0 \\ \sum_1^s a_i z^{-k_i} & 1 \end{pmatrix} \tag{3.3}$$

Remark 3.2. If $P(z)$ is a loop in $SL(2,\mathcal{F})$ the factorization involves a diagonal matrix of the type $D(m, n=-m)$. Hence the factorization is in terms of loops in $SL(2,\mathcal{F})$.

Remark 3.3. The above factorization is not of the Birkhoff type, where a loop is factorized into a term entire in z, a diagonal - monomial and a third term which is entire in $1/z$. It is, however, possible to derive a Birkhoff factorization from the theorem above, since in the statement of the theorem T can be replaced by S and the identity:

$$SAS^{-1} = S^{-1}AS \tag{3.4}$$

holds for any 2×2 matrix A. Hence all entire terms can be shuffled to the left and the terms entire in z^{-1} to the right.

The proof of Theorem 3.1 is based upon the following technical result:

Lemma 3.1. *Consider $P(z)\in X_m$ and denote by $P_e(z)$ and $P_o(z)$ the $\mathcal{F}^2$-valued polynomials: $P_e(z) = \sum_{n_{e,s}}^{n_{e,f}} P_{e,n} z^{-n}$ and $P_o(z) = \sum_{n_{o,s}}^{n_{o,f}} P_{o,n} z^{-n}$ such that $P(z) = \begin{pmatrix} P_e(z) \\ P_o(z) \end{pmatrix}$ and all four vectors $P_{e,n_{e,s}}, P_{e,n_{e,f}}, P_{o,n_{e,s}}, P_{o,n_{o,f}}$ are non-vanishing vectors in $\mathcal{F}^2$. By construction the determinant of P is of the form:*

$$detP(z) = \Delta_s z^{-(n_{e,s}+n_{o,s})} + \cdots + \Delta_f z^{(n_{e,f}+n_{o,f})} \tag{3.5}$$

with $\Delta_s = det\begin{pmatrix} P_{e,n_{e,s}} \\ P_{o,n_{o,s}} \end{pmatrix}$ and $\Delta_f = det\begin{pmatrix} P_{e,n_{e,f}} \\ P_{o,n_{o,f}} \end{pmatrix}$. Now, either of the three cases holds (exclusively):

Case 1: $\Delta_s = \Delta_f = 0$

Case 2: $(\mid \Delta_s \mid = 1$ *and* $\Delta_f = 0)$ *or* $(\Delta_s = 0$ *and* $\mid \Delta_f \mid = 1)$

Case 3: $n_{e,s} = n_{e,f}, n_{o,s} = n_{o,f}, n_{e,s} + n_{o,s} = m$

Proof. 1. det$P(z)$ is a Laurent polynomial in z, where Δ_s and Δ_f are the coefficients with the smallest and largest power of z^{-1}, respectively.

2. If $(n_{e,s} + n_{o,s}) < m < (n_{e,f} + n_{o,f})$ both coefficients have to vanish.

3. If $(n_{e,s} + n_{o,s}) = m < (n_{e,f} + n_{o,f})$ we find $\mid \Delta_s \mid = 1$ and $\Delta_f = 0$.

4. If $(n_{e,s} + n_{o,s}) = m = (n_{e,f} + n_{o,f})$ we argue as follows: $n_{e,s}$ and $n_{o,s}$ as well as $n_{e,f}$ and $n_{o,f}$ are symmetrically arranged with respect to m. On the other hand, $n_{e,s} \leq n_{e,f}$ and $n_{os,} \leq n_{o,f}$. This is only consistent with $n_{e,s} = n_{e,f}$ and $n_{o,s} = n_{o,f}$.

□

Proof. To prove the theorem, consider a loop $P \in X_m$. If we are in case 3 of the previous Lemma, then P is up to an elementary diagonal loop and a number of modulus one, a matrix in $SL(2, \mathcal{F})$. Such a matrix can be factorized by standard means into upper and lower triagonal matrices. Hence, for this particular case the proof is finished.
Assume now that we are not in Case 3 of the previous Lemma. We will now describe a reduction step, wich reduces the length of the filter by one. In either of the two cases 1 or 2 at least one of the two determinants Δ_s and Δ_f vanishes. Say this is $\Delta_s = 0$. So, we start working at the start of the filter and note that either $n_{e,s} \geq n_{o,s}$ or $n_{e,s} \leq n_{o,s}$. If $n_{e,s} \geq n_{o,s}$, apply a lower left shift $T_-^{(n_{e,s}-n_{o,s})}$ by multiplying from the left. The even and the odd parts $P_e(n_{e,s})$ and $P_o(n_{o,s})$ of the new filter at $n_{e,s}$ - again denoted by the same symbol - are both not zero. Multiplying by an appropriate upper triangular matrix of the type $\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix}$ one can annihilate the even part of the filter at the position $n_{e,s}$. A shift to the right by $T_-^{-(n_{e,s}-n_{o,s})}$ produces a loop which has $n_{s,e}$ reduced by one. If $\Delta_f = 0$, we operate on the final end of the filter analogously. If $n_{e,s} \leq n_{o,s}$ we apply an upper left shift $T_+^{(n_{o,s}-n_{e,s})}$. Again, the new filter has $P_e(n_{e,s}) \neq 0$ and $P_o(n_{o,s}) \neq 0$. A lower triangular matrix annihilates the odd part. Multiplication by $T_-^{(n_{o,s}-n_{e,s})}$ produces a filter where n_{os} is reduced by one. If we encounter case 3 of the previous Lemma, we get one single term of the diagonal type times a matrix with a determinant of absolute value one. Hence we have shown that every loop can be factorized into elementary triagonal matrices and elementary diagonal loops. □

A consequence of Theorem 3.1 is that every loop in X can be written as a product of shifts, i.e. loops given by the shift operator and of constant loops. More precisely we have:

Corollary 3.1. *Consider the two subgroups of* X*:*

$$\begin{array}{ll} 1. & \mathcal{C} = \{C(z) = C_0 \in SL(2,\mathcal{F})\} \\ 2. & \mathcal{S} = \{S^m \mid m \in \mathbb{Z}\} \end{array} \tag{3.6}$$

Then X *is generated algebraically by* $\mathcal{C}$ *and* $\mathcal{S}$*, i.e. each* $P \in X$ *is of the form:*

$$P = \lambda S^m C_1 S^{n_1} C_2 S^{n_2} \dots C_l S^{n_l} C_{l+1} \tag{3.7}$$

where $n_i \in \{-1, 0, +1\}$*,* $l \in \mathbb{N}$*,* $m \in \mathbb{Z}$*, and* $\lambda \in \mathbb{T}$*; all factors* S *are to the left and the factors* S^{-1} *to the right.*

Remark 3.4. If $P \in \mathcal{L}_{pol}U(2) \subset X$ the C matrices can be chosen from $U(2)$. This corresponds to orthogonal filters. [5]

Remark 3.5. Note that we have used in the proof of Theorem 3.1 operations acting from the left on the rows of P only. This is an arbitrary restriction imposed to simplify the algorithm.

The full set of transformations which may be used in the reduction steps is then:

- Acting from the left:
  - $\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix}$ add a multiple of the second row to the first;
  - $\begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix}$ add a multiple of the first row to the second;
  - $\begin{pmatrix} z^{-1} & 0 \\ 0 & 1 \end{pmatrix}$ shift upper rows to the right;
  - $\begin{pmatrix} 1 & 0 \\ a & z^{-1} \end{pmatrix}$ shift lower rows to the right;
- Acting from the right:
  - $\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix}$ add a multiple of the first column to the second;

$\begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix}$ add a multiple of the second column to the first;

$\begin{pmatrix} z^{-1} & 0 \\ 0 & 1 \end{pmatrix}$ shift left columns to the right;

$\begin{pmatrix} 1 & 0 \\ 0 & z^{-1} \end{pmatrix}$ shift right columns to the right.

Remark 3.6. It is along the lines of the proof of Theorem 3.1, i.e. by applying reduction steps from the left, that we arrive at the complexity estimate (4.4) below. The procedure can be replaced by operations acting from the right on the columns of P. In this case $n_{e,s}$, $n_{o,s}$, $n_{e,f}$ and $n_{o,f}$ are to be replaced by $n_{h,s}$, $n_{g,s}$, $n_{h,f}$ and $n_{g,f}$. Since $n_{g,f} - n_{g,s} + 1$ is the length of g and similarly for h, this consideration gives rise to the estimate 4.5.

4 Computational Aspects and Examples

Factorization of filters is of great importance for reducing computational complexity of transformations. After computing a bound for complexity, we apply the algorithm to some examples. The algorithm is finally presented in pseudocode form.

4.1 Complexity

Computational complexity of the factorized wavelet transform is essentially determined by the number of triangular matrices in the factorization of the loop $P(z)$. This is because the upper and lower shifts affect only the indexing.

Each term in the factorization adds to the total complexity $C(P)$ according to the following rule:

$$C(triangular\, matrix) = 2 \qquad \text{(one multiplication and two sums)} \qquad (4.1)$$

$$C(B) = 6 \qquad \text{(four multiplications and one sum)} \qquad (4.2)$$

$$C(shift) = C(D) = 0 \qquad \text{(reindexing)} \qquad (4.3)$$

A rough estimate leads to the bound

$$C(P) \leq 2(n_{e,f} - n_{o,f}) + 2(n_{o,f} - n_{o,s}) + 6 \tag{4.4}$$

The proof of the estimate (4.4) follows the argument in the proof of Theorem 3.1. In the cases 1 and 2 the length of the even or odd part of the filter is reduced by one. Each step is a multiplication by a triangular matrix and a shift and therefore it adds two units to the complexity. In case 3 we are left with a matrix of determinant one up to shifts and multiplication by a phase. This operation consists of 4 multiplications and 2 sums. Hence the complexity of this step is 6. As mentioned before we have used operations from the left only. In case we take the opposite point of view and act from the right one gets an analogous estimate.

Denote by $l(h)$ and $l(g)$ the two smallest even numbers larger or equal to the lengths of the two filters. In the same manner as above, we get the estimate (see also Remark 3.5).

$$C(P) \leq l(h) + l(g) + 2 \tag{4.5}$$

4.2 Examples

We apply the algorithm to four examples.

Example 4.1.

We take the following polyphase matrix:

$$\begin{pmatrix} z^{-1} + 3 + z & z^{-2} + 5z^{-1} + 5 + z \\ 1 + z & z^{-1} + 3 + z \end{pmatrix}$$

The factorization is as follows: $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & z^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ z & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & z^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} =$
$= \begin{pmatrix} 1 & 1 + z^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 + z & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 + z^{-1} \\ 0 & 1 \end{pmatrix}$

Example 4.2.

A biorthogonal basis having the cubic B-spline as primal scaling function. The polyphase matrix is defined by:

$$\begin{pmatrix} \frac{1}{8}(z^{-1}+6+z) & \frac{1}{32}(-3z^{-2}-5z^{-1}-5-3z) \\ \frac{1}{2}(1+z) & \frac{1}{8}(-3z^{-1}+10-3z) \end{pmatrix}$$

The factorization of the cubic B-spline reads:

$$\begin{pmatrix} 1 & \frac{1}{4} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{1}{4}z^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ z & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & -\frac{3}{16}z^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2} & -\frac{3}{8} \\ 0 & 2 \end{pmatrix} \tag{4.6}$$

This is an instance of a filter pair with coefficients in the ring of dyadic rationals that can be decomposed into factors with coefficients that remain in this ring.

Example 4.3.

This example is from Vetterli and Le Gall [13]. As the two authors note, it cannot be factorized into simple pseudo-unitary terms. [2] It is, however, factorizable in the sense of Theorem 3.1.

The defining polyphase matrix is:

$$\begin{pmatrix} z^{-2}+3z^{-1}+2+z & -\frac{1}{14}z^{-1}-\frac{5}{14}+\frac{2}{7}z+\frac{1}{14}z^{2} \\ z^{-2}+2z^{-1}+3+z & -\frac{1}{14}z^{-1}-\frac{2}{7}+\frac{5}{14}z+\frac{1}{14}z^{2} \end{pmatrix}$$

The factorization of this filter is:

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -z & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ z^{-1} & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -4 & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{1}{7}z \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{1}{7} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -\frac{7}{2}z^{-1} & 1 \end{pmatrix} \begin{pmatrix} -2 & 0 \\ 0 & -\frac{1}{2} \end{pmatrix} =$$
$$= \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -z+z^{-1}-4 & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{1}{7}(1+z) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -\frac{7}{2}z^{-1} & 1 \end{pmatrix} \begin{pmatrix} -2 & 0 \\ 0 & -\frac{1}{2} \end{pmatrix}$$

Example 4.4.

[2] Pseudo-unitary means, that the matrix is a unitary matrix up to a scaling in the directions of the coordinate lines.

Here is a filter pair with coefficients in $\mathbb{Z}$ that can be decomposed into factors with coefficients again in $\mathbb{Z}$ with this technique:

$$\begin{pmatrix} 1+2z & 4 \\ -z^2 & 1-2z \end{pmatrix}$$
$$= \begin{pmatrix} z^{-1} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & z \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & z^{-1} \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z & 0 \\ 0 & 1 \end{pmatrix}$$
$$= \begin{pmatrix} 1 & -2z^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -z^2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2z^{-1} \\ 0 & 1 \end{pmatrix}$$

4.3 Pseudocode

$P(z) = \begin{pmatrix} \sum_{n_{e,s}}^{n_{e,f}} P_{e,n} z^{-n} \\ \sum_{n_{o,s}}^{n_{e,f}} P_{o,n} z^{-n} \end{pmatrix}$

`factorList` $= \{\}$

`assert:` $\det P(z) = \mu z^{-m}$

$\lambda = \mu^{1/2}$ v $P(z) = (-1)^m \lambda^{-1} T_+^{-m} P(z)$

`AppendToFactorList` $\{\lambda I, T_+^m\}$

`while` $(n_{e,f} - n_{e,s}) > 0$ `or` $(n_{o,f} - n_{o,s}) > 0$

`(* reduce the first row until it becomes shorter than the second *)`

  `while` $(n_{e,f} - n_{e,s}) \geq (n_{o,f} - n_{o,s})$ `and` $(n_{e,f} - n_{e,s}) > 0$

    `if` det $\begin{pmatrix} P_{e,n_{e,s}} \\ P_{o,n_{o,s}} \end{pmatrix} = 0$ `then`

      `ReduceSUpper()`

    `else`

```
      ReduceFUpper()

  end while

(* reduce the second row until it becomes shorter than the first *)

  while (n_{o,f} − n_{o,s}) ≥ (n_{e,f} − n_{e,s}) and (n_{o,f} − n_{o,s}) > 0
```

$$\text{if } \det\begin{pmatrix} P_{e,n_{e,s}} \\ P_{o,n_{o,s}} \end{pmatrix} = 0 \text{ then}$$

```
      ReduceSLower()

    else

      ReduceFLower()

  end while

end while

AppendToFactorList {P(z)}

begin subroutine ReduceSUpper()

  if h_{o,n_{o,s}} ≠ 0 then
```

$$U = \begin{pmatrix} 1 & -\frac{h_{e,n_{e,s}}}{h_{o,n_{o,s}}} \\ 0 & 1 \end{pmatrix}$$

```
  else
```

$$U = \begin{pmatrix} 1 & -\frac{g_{e,n_{e,s}}}{g_{o,n_{o,s}}} \\ 0 & 1 \end{pmatrix}$$

$$U^{-1} = \begin{pmatrix} 1 & -U_{12} \\ 0 & 1 \end{pmatrix}$$

AppendToFactorList $(\{T_+^{(n_{o,s}-n_{e,s})}\, U^{-1}\, T_+^{-(n_{o,s}-n_{e,s})}\})$

$P(z) = T_+^{-(n_{o,s}-n_{e,s})}\, U\, T_+^{(n_{o,s}-n_{e,s})}\, P(z)$

```
end ReduceSUpper()
```

```
begin subroutine ReduceSLower()
```

`if` $h_{e,n_{e,s}} \neq 0$ `then`

$$L = \begin{pmatrix} 1 & 0 \\ -\frac{h_{o,n_{o,s}}}{h_{e,n_{e,s}}} & 1 \end{pmatrix}$$

`else`

$$L = \begin{pmatrix} 1 & 0 \\ -\frac{g_{o,n_{o,s}}}{g_{o,n_{o,s}}} & 1 \end{pmatrix}$$

$$L^{-1} = \begin{pmatrix} 1 & 0 \\ -L_{21} & 1 \end{pmatrix}$$

`AppendToFactorList` $(\{T_-^{(n_{e,s}-n_{o,s})}\, L^{-1}\, T_-^{-(n_{e,s}-n_{o,s})}\})$

$P(z) = T_-^{-(n_{e,s}-n_{o,s})}\, L\, T_-^{(n_{e,s}-n_{o,s})}\, P(z)$

```
end ReduceSLower()
```

```
begin subroutine ReduceFUpper()
```

`if` $h_{o,n_{o,f}} \neq 0$ `then`

$$U = \begin{pmatrix} 1 & -\frac{h_{e,n_{e,f}}}{h_{o,n_{o,f}}} \\ 0 & 1 \end{pmatrix}$$

`else`

$$U = \begin{pmatrix} 1 & -\frac{g_{e,n_{e,f}}}{g_{o,n_{o,f}}} \\ 0 & 1 \end{pmatrix}$$

$$U^{-1} = \begin{pmatrix} 1 & -U_{12} \\ 0 & 1 \end{pmatrix}$$

`AppendToFactorList` $(\{T_+^{(n_{o,f}-n_{e,f})}\, U^{-1}\, T_+^{-(n_{o,f}-n_{e,f})}\})$

$$P(z) = T_+^{-(n_{o,f}-n_{e,f})}\, U\, T_+^{(n_{o,f}-n_{e,f})}\, P(z)$$

`end ReduceFUpper()`

`begin subroutine ReduceELower()`

`if` $h_{e,n_{e,f}} \neq 0$ `then`

$$U = \begin{pmatrix} 1 & 0 \\ -\frac{h_{o,n_{o,f}}}{h_{e,n_{e,f}}} & 1 \end{pmatrix}$$

`else`

$$L = \begin{pmatrix} 1 & 0 \\ -\frac{g_{o,n_{o,f}}}{g_{e,n_{e,f}}} & 1 \end{pmatrix}$$

$$L^{-1} = \begin{pmatrix} 1 & 0 \\ -U_{21} & 1 \end{pmatrix}$$

`AppendToFactorList` $(\{T_-^{(n_{e,f}-n_{o,f})}\, L^{-1}\, T_-^{-(n_{e,f}-n_{o,f})}\})$

$$P(z) = T_-^{-(n_{e,f}-n_{o,f})}\, L\, T_-^{(n_{e,f}-n_{o,f})}\, P(z)$$

`end ReduceFLower()`

References

[1] I. Daubechies. Ten Lectures on Wavelets. *CBMS-NSF Regional Conf. Series in Appl. Math.*, 61, 1992.

[2] I. Daubechies and W. Sweldens. Factoring Wavelet Transform into Lifting Steps. *Technical Report*, 1996, 1996.

[3] F. Dorfmeister, J. , Pedit and H. Wu. Weierstrass type reprsentation of harmonic maps into symmetric spaces. *Communications in Analysis and Geometry*, to appear.

[4] L.D. Faddeev and L.A. Takhtajan. *Hamiltonian Methods in the Theory of Solitons*. Springer Verlag, 1987.

[5] M. Holschneider and U. Pinkall. Quadrature Mirror Filters and Loop Groups. *SFB-Preprint, TU-Berlin*, 1993.

[6] A. Logar and B. Sturmfels. Algorithms for the quillen-shuslin theorem. *Journal of Algebra*, 145:231–239, 1992.

[7] T.Y. Nguyen and P.P. Vaidyanathan. Perfect reconstruction QMF structures which yield linear, phase FIR analysis filters. In *Proceedings of the Intenational Symposium on Circuits Systems*, pages 297–300. 1988.

[8] H Park and C. Woodburn. An algorithmic proof of Shuslin's stability theorem for polynomial rings. *Journal of Algebra*, to appear, 1995.

[9] H-J Park. A computational theory of laurent polynomial rings and multidimensional fir systems. *PhD Thesis, Dept.of Mathematics, University of Berkeley*, 1995.

[10] A. Pressley and G. Segal. *Loop Groups*. Oxford Math. Monographs. Clarendon Press, 1986.

[11] H. Tolhuizen, L., Hollmann and A. Kalker. On the realizability of biorthogonal m-dimensional 2-band filter banks. 1995.

[12] M. Vetterli. Filter banks allowing perfect reconstruction. *Signal Process*, 10:219–244, 1986.

[13] M. Vetterli and D. Le Gall. Perfect reconstruction FIR filter banks: Some properties and factorizations. *IEEE Trans. Acoust. Speech Signal Process*, 37:1057–1071, 1989.