Multiwavelet frames in signal space originated from Hermite splines*

Amir Z. Averbuch*, Valery A. Zheludev, and Tamir Cohen
{amir1,zhel,tamircoh}@post.tau.ac.il
School of Computer Science
Tel Aviv University
Tel Aviv 69978,Israel

Abstract

We present a method for construction of multiwavelet frames for manipulation of discrete signals. The frames are generated by three-channel perfect reconstruction oversampled multifilter banks. The design of the multifilter bank starts from a pair of interpolatory multifilters. We derive these interpolatory multifilters from the cubic Hermite splines. We use original pre-processing algorithms, which transform scalar signals into vector arrays that serve as inputs to the oversampled analysis multifilter banks. These pre-processing algorithms do not degrade the approximation accuracy of the transforms of the vectors by multifilter banks. The post-processing algorithms convert the vector output of the synthesis multifilter banks into scalar signal. The discrete framelets, generated by the designed filter banks, are symmetric and have short support. The analysis framelets have four vanishing moments, whereas the synthesis framelets converge to Hermite splines supported on the interval [-1,1].

1 Introduction

As wavelet analysis grows into a well-developed and widely applied theory, much research effort have focused on multiwavelets and, more recently, on wavelet frames also called framelets. Multiwavelet transforms, whose investigation was initiated in [1, 2, 16, 18, 20], are associated with several scaling functions. A basis for $L^2(\mathbb{R})$ is formed by translations and dilations of several generating wavelets. Such a scheme provides more flexibility in design and applications compared to the conventional wavelet transforms. In particular, it is possible to achieve certain approximation accuracy using waveforms with shorter support than with standard wavelet analysis. Once appropriate coding schemes are available, multiwavelet transforms have a strong potential to achieve good image compression [9, 15]. Wavelet frames correspond to a single scaling function and several wavelets and provide redundant expansions of functions. Construction methods with extensive investigation of the properties of wavelet frames were presented in [7, 8, 24, 21, 13, 12, 22, 23]. A natural idea is to construct frames that correspond to multiwavelet transforms. To the authors knowledge, by now only [14] presents a detailed investigation in this direction. There a general theory of multiwavelet frames in L^2 is developed and an algorithm for the construction of these frames have been presented. The construction algorithm in [14] starts from any two refinable vector functions.

Unlike [14], in this paper we focuse on multiwavelet frames in the signal space l_2 rather than in the functional space L^2. Oversampled filter banks provide a natural source to devise wavelet-type frames in the signal space [6, 10, 4]. In the multiwavelet case, the filter banks consist of matrix filters, the so-called multifilters. We discuss three-channel multifilter banks with downsampling factor $N = 2$. They produce frames with minimal redundancy. We propose a generic method to design a perfect reconstruction multifilter bank starting with an arbitrary interpolatory "low-pass"

*The research was partially supported by STRIMM Consortium (2003-2005) administrated by the Ministry of Trade of Israel.

multifilter. For the design of such a multifilter, we use properties of the interpolatory cubic Hermite splines. These splines are frequently used for multiwavelet constructions [18, 20, 25, 26, 19, 3]. They produce multifilters with a favorable combination of properties: short impulse response, linear phase and good approximation accuracy. The coefficients of the filters are integers that accelerate the implementation. However, different interpolating aggregates can be used for the presented scheme.

To exploit the useful properties of the multifilter bank originating from the cubic Hermite splines, the input to the analysis multifilter bank must be a vector array with a special structure. To produce such an array from the original scalar signal, a pre-processing procedure is required. This procedure can be organized as filtering the original signal by a critically sampled two-channel filter bank. This vector array serves as an input to the main oversampled analysis multifilter bank, which produces three vector streams. For the multiscale processing, the latter operation is iterated using the vector array produced by the "low-pass" multifilter as an input . The synthesis procedure consists of application of the synthesis multifilter bank to the three vector streams, which creates a vector array. A post-processing filter bank, which is critically sampled, transforms this vector array into a scalar signal. We present three different pre(post)-processing filter banks, which were introduced in our previous paper [3] that was devoted to the design of multiwavelet bases in signal space. The filters in the filter banks are linear phase, have short impulse response and retain the approximation accuracy inherent in the main block-wise filter banks. Filtering can be implemented in a fast lifting mode. The combination of the pre(post)- processing filter bank with the analysis(synthesis) multifilter bank forms Multiple Input Multiple Output (MIMO) filter banks.

The filter banks produce frames in the signal space that consist of translations of several compactly supported generating signals, which we call discrete framelets. Usage of Hermite splines enables us to devise framelets with useful properties for signal processing. All are symmetric or antisymmetric, with very short support. The "high-frequency" analysis framelets have four vanishing moments. The synthesis framelets converge in the sense of the "cascade algorithm" [11] to Hermite splines, which are supported on the interval [-1 1] and have explicit representation. It is interesting to point out that the limit functions do not depend on the employed post-processing filter bank.

The paper is organized as follows. In Section 2 we outline the necessary facts on multifilter banks and pre-(post)processing filter banks and explain their relation to the frames in signal space. Section 3, which is central in the paper, is devoted to the design of perfect reconstruction multifilter banks with interpolatory multifilters. In addition, we devise multifilter banks that are based on Hermite splines. In Section 4 we present three pre-(post)processing filter banks that are adjusted to the above multifilter banks. In Section 5 we establish some properties of the analysis and synthesis framelets, such as symmetry and vanishing moments. We prove that the discrete synthesis framelets converge to Hermite splines, which are supported on the interval [-1 1]. In Section 6 we display the devised framelets.

2 Multifilter banks and frames

2.1 Multifilter banks

Let $\mathbf{H} = \{H(l)\}_{l\in\mathbb{Z}}$ be the matrix sequence:

$$H(l) \stackrel{\Delta}{=} \begin{pmatrix} h_{11}(l) & h_{12}(l) \\ h_{21}(l) & h_{22}(l) \end{pmatrix}, \quad l \in \mathbb{Z},$$

and $\vec{x} \stackrel{\Delta}{=} \{\vec{x}(l)\} = (x_1(l), x_2(l))^T,\ l \in \mathbb{Z}$, be a vector signal. The following operation

$$\vec{y} = \mathbf{H} * \vec{x} \Longleftrightarrow \vec{y}(r) = \sum_l H(r-l)\vec{x}(l)$$

is called the matrix-vector convolution or multifiltering the signal $\vec{x}$. The matrix sequence $\mathbf{H} = \{H(l)\}_{l\in\mathbb{Z}}$ is called multifilter.

The $z-$transform of the vector signal $\vec{x} = \{\vec{x}(l)\}$ and of the multifilter $\mathbf{H} = \{H(l)\}_{l\in\mathbb{Z}}$ are:

$$\vec{X}(z) \triangleq \sum_{l\in\mathbb{Z}} z^{-l}\vec{x}(l),\ \mathbf{H}(z) \triangleq \sum_{l\in\mathbb{Z}} z^{-l}H(l) = \begin{pmatrix} h_{11}(z) & h_{12}(z) \\ h_{21}(z) & h_{22}(z) \end{pmatrix},\ \vec{y} = \mathbf{H} * \vec{x} \Longleftrightarrow \vec{Y}(z) = \mathbf{H}(z)\vec{X}(z).$$

Throughout the paper we assume that $z = e^{j\omega}$ and all the multifilters components $\{H(l)\}$ are real-valued. We designate a multifilter by its z-transform.

In this paper, we consider only 3-channel multifilter banks with downsampling factor $N = 2$. The analysis multifilter bank consists of the multifilters $\tilde{\mathbf{H}}^k(z)$ and the synthesis multifilter bank consists of the multifilters $\mathbf{H}^k(z),\ k = 0, 1, 2$. We denote the output signals from the analysis multifilter bank by $\vec{y}^{k,1},\ k = 0, 1, 2$. These signals are used as the input to the synthesis multifilter bank. Then, the analysis and synthesis formulas become

$$\vec{y}^{k,1}(l) = \sum_{n\in\mathbb{Z}} \tilde{H}^k(n-2l)\,\vec{x}(n) \Leftrightarrow \vec{Y}^{k,1}(z^2) = \frac{\tilde{\mathbf{H}}^k(1/z)\vec{X}(z) + \tilde{\mathbf{H}}^k(-1/z)\vec{X}(-z)}{2},\ k = 0, 1.2, \tag{1}$$

$$\hat{\vec{x}}(l) = \sum_{k=0}^{2}\sum_{n\in\mathbb{Z}} H^k(l-2n)\,\vec{y}^{k,1}(n) \Leftrightarrow \hat{\vec{X}}(z) = \sum_{k=0}^{2}\mathbf{H}^k(z)\vec{Y}^{k,1}(z^2). \tag{2}$$

Remark The signals $\vec{y}^{k,1}$ have double superscript indices $\{k, 1\}$, where k corresponds to the index of the analysis multifilter $\tilde{\mathbf{H}}^k$, which produces the signal. The second index 1 means the first decomposition scale. The multiscale transforms will be introduced in Section 2.4.

2.1.1 Polyphase representation

The z-transforms of a vector signal $\vec{x}$ and a multifilter $\mathbf{H}$ can be represented in the following polyphase mode

$$\vec{X}(z) = \vec{E}(z^2) + z^{-1}\vec{O}(z^2), \quad \mathbf{H}(z) = \mathbf{H}_e(z^2) + z^{-1}\mathbf{H}_o(z^2),$$

where

$$\vec{E}(z) \triangleq \sum_{k\in\mathbb{Z}} z^{-k}\vec{e}(k), \quad \vec{O}(z) \triangleq \sum_{k\in\mathbb{Z}} z^{-k}\vec{o}(k), \quad \vec{e}(k) \triangleq \vec{x}(2k),\ \vec{o}(k) \triangleq \vec{x}(2k+1),$$
$$\mathbf{H}_e(z) \triangleq \sum_{k\in\mathbb{Z}} z^{-k}H(2k), \quad \mathbf{H}_o(z) \triangleq \sum_{k\in\mathbb{Z}} z^{-k}H(2k+1).$$

Application of the time-reversed multifilter $\mathbf{H}$ to the signal $\vec{x}$ is represented by

$$\begin{aligned} \vec{g}(r) = \textstyle\sum_k H(k-r)\vec{x}(k) \Leftrightarrow \qquad & \vec{G}(z) = \mathbf{H}(1/z)\vec{X}(z) = \vec{G}_e(z^2) + z^{-1}\vec{G}_o(z^2), \\ \vec{G}_e(z) = \mathbf{H}_e(1/z)\vec{E}(z) + \mathbf{H}_o(1/z)\vec{O}(z), \qquad & \vec{G}_o(z) = \mathbf{H}_e(1/z)\vec{O}(z) + z\mathbf{H}_o(1/z)\vec{E}(z). \end{aligned} \tag{3}$$

Thus, the analysis and synthesis Eqs. (1), (2), respectively, can be rewritten as

$$\begin{pmatrix} \hat{\vec{E}}(z) \\ \hat{\vec{O}}(z) \end{pmatrix} = \mathbf{P}(\mathbf{z}) \cdot \begin{pmatrix} \vec{Y}^0(z) \\ \vec{Y}^1(z) \\ \vec{Y}^2(z) \end{pmatrix}, \quad \begin{pmatrix} \vec{Y}^0(z) \\ \vec{Y}^1(z) \\ \vec{Y}^2(z) \end{pmatrix} = \tilde{\mathbf{P}}(1/z) \cdot \begin{pmatrix} \vec{E}(z) \\ \vec{O}(z) \end{pmatrix},$$

where

$$\tilde{\mathbf{P}}(z) \triangleq \begin{pmatrix} \tilde{\mathbf{H}}_e^0(z) & \tilde{\mathbf{H}}_o^0(z) \\ \tilde{\mathbf{H}}_e^1(z) & \tilde{\mathbf{H}}_o^1(z) \\ \tilde{\mathbf{H}}_e^2(z) & \tilde{\mathbf{H}}_o^2(z) \end{pmatrix}, \quad \mathbf{P}(z) \triangleq \begin{pmatrix} \mathbf{H}_e^0(z) & \mathbf{H}_e^1(z) & \mathbf{H}_e^2(z) \\ \mathbf{H}_o^0(z) & \mathbf{H}_o^1(z) & \mathbf{H}_o^2(z) \end{pmatrix} \tag{4}$$

are the analysis $\tilde{\mathbf{P}}(z)$ and synthesis $\mathbf{P}(z)$ block-wise *polyphase matrices* .

If the output vector signal $\hat{\vec{x}}$ is equal to the input signal $\vec{x}$ then the analysis and synthesis multifilter banks form a perfect reconstruction multifilter bank. Analytically, this property is expressed via the polyphase matrices as:

$$\mathbf{P}(z) \cdot \tilde{\mathbf{P}}(1/z) = \begin{pmatrix} \mathbf{I_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{I_2} \end{pmatrix}, \tag{5}$$

where $\mathbf{I_2}$ is the 2×2 identity matrix. Thus, the synthesis polyphase matrix has to be a left inverse of the analysis matrix. Obviously, if such a matrix exists, it is not unique.

A necessary condition for (5) to hold is that the analysis matrix $\tilde{\mathbf{P}}(z)$ has a full rank of 4 on the unit circle $|z| = 1$. However, if this condition is satisfied then there exists at least one solution to (5), namely, the parapseudoinverse of $\tilde{\mathbf{P}}$:

$$\mathbf{P}(z) = \tilde{\mathbf{P}}^{+}(z) \triangleq \left(\tilde{\mathbf{P}}^{*}(z) \cdot \tilde{\mathbf{P}}(z)\right)^{-1} \cdot \tilde{\mathbf{P}}^{*}(z).$$

Here we use the common notation $H^{*}(z) \triangleq H(1/z)^{T}$.

2.2 Pre-(post)processing filter banks

The goal of the pre-processing is to generate from the scalar signal $f(k)$ the vector signal $\vec{x}(k) \triangleq (x_1(k), x_2(k))^T$, $k \in \mathbb{Z}$, with some desired properties, which serves as an input to the analysis multifilter bank. The post-processing procedure has to restore the scalar signal from the vector, which is the output from the synthesis multifilter bank.

A straightforward way to convert a scalar signal $f(k)$ into a vector array is to use its polyphase components as the coordinates of the vector $\vec{x}(k) = \vec{t}(k) \triangleq (f_e(k), f_o(k))^T \Leftrightarrow \vec{X}(z) = \vec{T}(z)$, where $f_e(k) \triangleq f(2k)$, $f_o(k) \triangleq f(2k+1)$ and $\vec{T}(z) \triangleq (F_e(z,)\ F_o(z))^T$. However, in order for the vector array $\vec{x}(k)$ to have necessary properties, the array $\vec{t}(k)$ is subjected to the transform:

$$\vec{X}(z) = \tilde{\mathbf{R}}(1/z) \cdot \vec{T}(z), \quad \vec{T}(z) = \mathbf{R}(z) \cdot \vec{X}(z), \text{ where} \tag{6}$$

$$\tilde{\mathbf{R}}(z) \triangleq \begin{pmatrix} \tilde{R}_e^1(z) & \tilde{R}_o^1(z) \\ \tilde{R}_e^2(z) & \tilde{R}_o^2(z) \end{pmatrix}, \quad \mathbf{R}(z) \triangleq \begin{pmatrix} R_e^1(z) & R_e^2(z) \\ R_o^1(z) & R_o^2(z) \end{pmatrix} = \tilde{\mathbf{R}}(1/z)^{-1}.$$

These procedures are equivalent to processing by two-channel critically sampled filter banks. As usual, $F(z)$ means the z-transform of the signal $f(k)$. The filter banks, whose polyphase matrices are $\tilde{R}^r(z)$ and $R^r(z)$, are defined as:

$$\tilde{R}^r(z) = \tilde{R}_e^r(z^2) + z^{-1}\tilde{R}_o^r(z^2), \quad R^r(z) = R_e^r(z^2) + z^{-1}R_o^r(z^2), \ r = 1, 2. \tag{7}$$

Then

$$X_r(z^2) = \frac{\tilde{R}^r(1/z)F(z) + \tilde{R}^r(-1/z)F(-z)}{2}, \ r = 1, 2, \quad F(z) = \sum_{r=1}^{2} R^r(z) X_r(z^2). \tag{8}$$

The polyphase components of $\vec{X}(z)$ are

$$\vec{E}(z) = \tilde{\mathbf{R}}_e(1/z)\vec{T}_e(z) + \tilde{\mathbf{R}}_o(1/z)\vec{T}_o(z), \ \vec{O}(z) = \tilde{\mathbf{R}}_e(1/z)\vec{T}_o(z) + z\tilde{\mathbf{R}}_o(1/z)\vec{T}_e(z). \tag{9}$$

Here, $\tilde{\mathbf{R}}_e(1/z)$, $\tilde{\mathbf{R}}_o(1/z)$, $\vec{T}_e(z)$ and $\vec{T}_o(z)$ are the polyphase components of $\tilde{\mathbf{R}}(1/z)$ and $\vec{T}(z)$, respectively:

$$\tilde{\mathbf{R}}_e(1/z) = \sum_{k\in\mathbb{Z}} z^k \tilde{R}(2k), \quad \tilde{\mathbf{R}}_o(1/z) = \sum_{k\in\mathbb{Z}} z^k \tilde{R}(2k+1),$$

$$\vec{T}_e(z) = \sum_{k\in\mathbb{Z}} z^{-k} \begin{pmatrix} f(4k) \\ f(4k+1) \end{pmatrix}, \quad \vec{T}_o(z) = \sum_{k\in\mathbb{Z}} z^{-k} \begin{pmatrix} f(4k+2) \\ f(4k+3) \end{pmatrix}.$$

The four-fold polyphase components of the signal $f(n)$ are

$$F_n(z) \stackrel{\Delta}{=} \sum_{k\in\mathbb{Z}} z^{-k} f(4k+n),\ n=0,1,2,3, \quad \vec{F}^p(z) \stackrel{\Delta}{=} (F_0(z),F_1(z),F_2(z),F_3(z))^T .$$

The block-wise 4×4 polyphase matrices are defined to be

$$\tilde{\mathbf{R}}^p(z) \stackrel{\Delta}{=} \begin{pmatrix} \tilde{\mathbf{R}}_e(z) & \tilde{\mathbf{R}}_o(z) \\ z^{-1}\tilde{\mathbf{R}}_o(z) & \tilde{\mathbf{R}}_e(z) \end{pmatrix}, \quad \mathbf{R}^p(z) \stackrel{\Delta}{=} \begin{pmatrix} \mathbf{R}_e(z) & z^{-1}\mathbf{R}_o(z) \\ \mathbf{R}_o(z) & \mathbf{R}_e(z) \end{pmatrix} = \tilde{\mathbf{R}}^p(1/z)^{-1}.$$

Then, due to (9), the pre-processing procedure can be represented as

$$\begin{pmatrix} \vec{E}(z) \\ \vec{O}(z) \end{pmatrix} = \tilde{\mathbf{R}}^p(1/z)\cdot \vec{F}^p(z). \tag{10}$$

Applying the analysis multifilter bank to the output from the pre-processing filter bank, we obtain

$$\begin{pmatrix} \vec{Y}^0(z) \\ \vec{Y}^1(z) \\ \vec{Y}^2(z) \end{pmatrix} = \tilde{\mathbf{P}}(1/z)\cdot \begin{pmatrix} \vec{E}(z) \\ \vec{O}(z) \end{pmatrix} = \tilde{\mathbf{S}}(1/z)\cdot \vec{F}^p(z),$$

where

$$\tilde{\mathbf{S}}(z) \stackrel{\Delta}{=} \tilde{\mathbf{P}}(z)\cdot \tilde{\mathbf{R}}^p(z) \tag{11}$$

is a 6×4 polyphase matrix of the oversampled MIMO filter bank that transforms the scalar signal $f(k)$ into the vector arrays of the coefficients $\vec{Y}^r(k),\ r=0,1,2$. The transform is illustrated by the diagram in Fig. 1. The synthesis procedure can be represented as

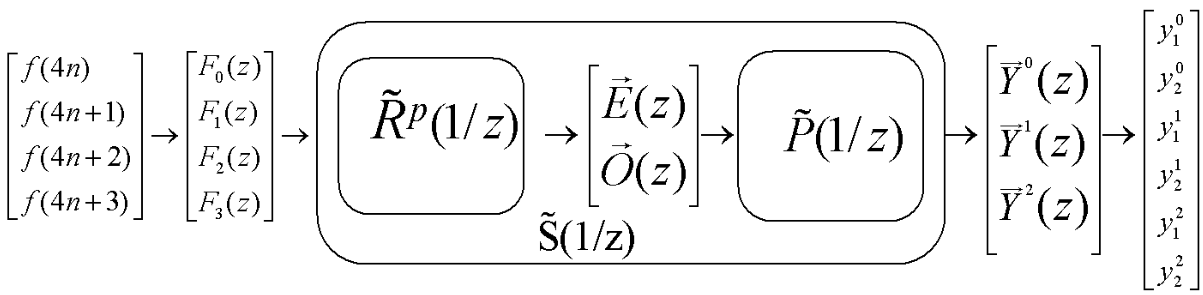

Figure 1: Transform of the scalar signal $f(k)$ into arrays of coefficients by MIMO analysis filter bank with the polyphase matrix $\tilde{\mathbf{S}}(z)$.

$$\vec{T}(z) = \mathbf{R}(z)\cdot \vec{X}(z) \Longleftrightarrow \vec{F}^p(z) = \mathbf{R}^p(z)\cdot \begin{pmatrix} \vec{E}(z) \\ \vec{O}(z) \end{pmatrix} = \mathbf{S}(z)\cdot \begin{pmatrix} \vec{Y}^0(z) \\ \vec{Y}^1(z) \\ \vec{Y}^2(z) \end{pmatrix},$$

where

$$\mathbf{S}(z) \stackrel{\Delta}{=} \mathbf{R}^p(z)\cdot \mathbf{P}(z), \tag{12}$$

is a 4×6 polyphase matrix of the oversampled MIMO filter bank that transforms arrays of the coefficients $\vec{Y}^r(k),\ r=0,1,2,$ into the polyphase representation of the scalar signal $f(k)$. This transform is illustrated by the diagram in Fig. 2.

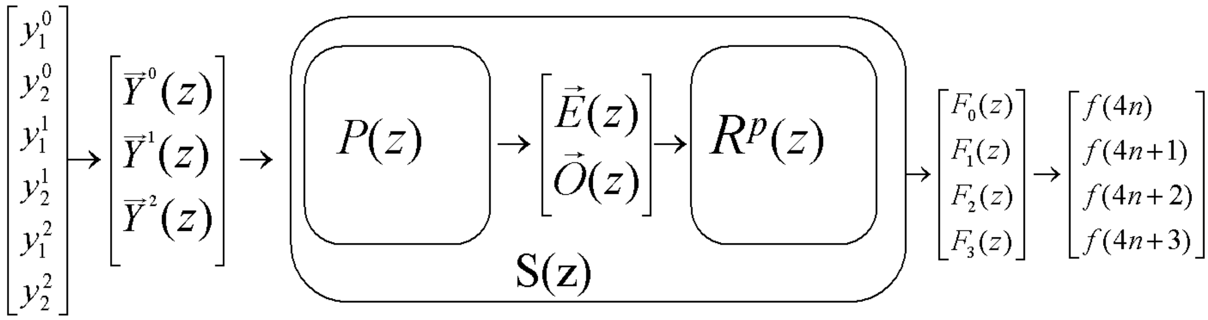

Figure 2: Transform of arrays of coefficients into the scalar signal $f(k)$ by MIMO synthesis filter bank with the polyphase matrix $\mathbf{S}(z)$.

2.3 Frames

In the rest of the paper we use the following notation. If $f(k)$ and $g(k)$ belong to the space l_2 of square summable signals then $\langle f, g\rangle \stackrel{\Delta}{=} \sum_{k\in\mathbb{Z}} f(k)g(k)$ is the inner product in l_2. If $\vec{a} = (a_1, a_2)^T$ and $\vec{b} = (b_1, b_2)^T$ then $\langle \vec{a}, \vec{b}\rangle_2 \stackrel{\Delta}{=} a_1b_1 + a_2b_2$ is the scalar product in $\mathbb{R}^2$. We use the notation

$$[\vec{a}, g] \stackrel{\Delta}{=} \begin{pmatrix} \langle a_1, g\rangle \\ \langle a_2, g\rangle \end{pmatrix}$$

for the "inner product" of the vector signal $\vec{a}(l)$ with the scalar signal $g(l)$.

Definition 2.1 *A system $\tilde{\Phi} \stackrel{\Delta}{=} \{\tilde{\phi}_j\}_{j\in\mathbb{Z}}$ of signals forms a frame in the signal space l_2 if there exist positive constants A and B such that for any signal $f \in l_2$*

$$A\|f\|^2 \le \sum_{j\in\mathbb{Z}} |\langle f, \tilde{\phi}_j\rangle|^2 \le B\|f\|^2.$$

If a system $\tilde{\Phi}$ is a frame then there exists another frame $\Phi \stackrel{\Delta}{=} \{\phi_i\}_{i\in\mathbb{Z}}$ in the signals space such that any signal f can be expanded into the sum $f = \sum_{i\in\mathbb{Z}} \langle f, \tilde{\phi}_i\rangle\, \phi_i$. The frames $\tilde{\Phi}$ and Φ can be interchanged. Together they form the so-called bi-frame.

Following is the condition for the analysis filter bank to provide a frame expansion in the signal space (Cvetković and Vetterli [10]).

Proposition 2.1 ([10]) *A FIR analysis filter bank provides a frame expansion of the signals $f \in l_2$ if and only if its polyphase matrix is of full rank on the unit circle $|z| = 1$.*

Corollary 2.1 ([10]) *A FIR analysis filter bank with the polyphase matrix $\tilde{\mathbf{G}}(\mathbf{z})$ implements a frame expansion of signals $f \in l_2$ if and only if there exists a matrix $\mathbf{G}(\mathbf{z})$ of stable rational functions*[1]*, such that $\mathbf{G}(\mathbf{z})\tilde{\mathbf{G}}(\mathbf{1/z}) = c\mathbf{I}$.*

Corollary 2.2 *A perfect reconstruction filter bank, whose analysis polyphase matrix consists of Laurent polynomials and the synthesis polyphase matrix consists of stable rational functions produces a bi-frame of the signal space l_2.*

We describe the structure of the frame expansion in our setting.

[1]rational functions that do not have poles on the unit circle $|z| = 1$.

If the rank of the pre-processing polyphase matrix $\tilde{\mathbf{R}}(z)$ is equal to 2 then there exists the post-processing matrix $\mathbf{R}(z) = \tilde{\mathbf{R}}^{-1}(1/z)$ and the pre-post processing formulas can be represented by a biorthogonal basis expansion.

Denote by $\{\varphi_1^0(k)\}$ and $\{\varphi_2^0(k)\}$, $k \in \mathbb{Z}$, the impulse response of the filters $R^1(z)$ and $R^2(z)$, respectively, and let $\vec{\varphi}^0(k) \triangleq \left(\varphi_1^0(k), \varphi_2^0(k)\right)^T$. Eq. (8) implies the following representation of the signal f:

$$f(k) = \sum_{l\in\mathbb{Z}} x_1(l)\varphi_1^0(k-2l) + \sum_{l\in\mathbb{Z}} x_2(l)\varphi_2^0(k-2l) = \sum_{l\in\mathbb{Z}} \langle \vec{x}(l), \vec{\varphi}^0(k-2l)\rangle_2. \tag{13}$$

From (8) we have

$$X_1(z) = \left(\tilde{R}^1(1/z)F(z)\right)_e, \quad X_2(z) = \left(\tilde{R}^2(1/z)F(z)\right)_e, \tag{14}$$

where the subscript $\cdot_e$ means the even polyphase component. Denote by $\{\tilde{\varphi}_1^0(k)\}$ and $\{\tilde{\varphi}_2^0(k)\}$, $k \in \mathbb{Z}$, the impulse response of the filters $\tilde{R}^1(z)$ and $\tilde{R}^2(z)$, respectively, and let $\tilde{\vec{\varphi}}^0(k) \triangleq \left(\tilde{\varphi}_1^0(k), \tilde{\varphi}_2^0(k)\right)^T$. Then, Eq. (14) can be rewritten as

$$\vec{x}(k) = \sum_{l\in\mathbb{Z}} \tilde{\vec{\varphi}}^0(l-2k)f(l) = \left[\tilde{\vec{\varphi}}^0(\cdot - 2k), f\right]. \tag{15}$$

Remark. If the rank of the pre-processing polyphase matrix $\tilde{\mathbf{R}}(z)$ is equal to 2 then the relation $\vec{X}(z) = \tilde{\mathbf{R}}(1/z) \cdot \vec{T}(z)$ is bijective. Consequently, the relation (10) is bijective as well. Hence, the rank of the matrix $\tilde{\mathbf{R}}^p(z)$ is equal to four. If the matrix $\tilde{\mathbf{P}}(z)$, defined in (4), has a full rank 4 then the analysis polyphase matrix $\tilde{\mathbf{S}}(z)$, defined in (11), has a full rank four. The corresponding analysis filter bank provides a frame expansion of signals and there exists the synthesis filter bank whose polyphase matrix $\mathbf{S}(z)$ is given by (12).

Let $\tilde{H}^r(l)$ and $H^r(l)$, $l \in \mathbb{Z}$, $r = 0,1,2$, be the impulse response of the multifilters $\tilde{\mathbf{H}}^r(z)$ and $\mathbf{H}^r(z)$, respectively. Equations (13) and (2) imply that

$$\begin{aligned} f(k) &= \sum_{l\in\mathbb{Z}} \langle \vec{x}(l), \vec{\varphi}^0(k-2l)\rangle_2 = \sum_{l\in\mathbb{Z}} \left\langle \sum_{r=0}^{2}\sum_{n\in\mathbb{Z}} H^r(l-2n)\, \vec{y}^{r,1}(n), \vec{\varphi}^0(k-2l)\right\rangle_2 \\ &= \sum_{r=0}^{2}\sum_{n\in\mathbb{Z}} \left\langle \vec{y}^{r,1}(n)\,, \sum_{l\in\mathbb{Z}} H^r(l-2n)^T\, \vec{\varphi}^0(k-2l)\right\rangle_2 = \sum_{r=0}^{2}\sum_{n\in\mathbb{Z}} \left\langle \vec{y}^{r,1}(n)\,, \vec{\varphi}^{r,1}(k-4n)\right\rangle_2, \end{aligned}$$

where

$$\vec{\varphi}^{r,1}(k) \triangleq \sum_{l\in\mathbb{Z}} H^r(l)^T \vec{\varphi}^0(k-2l) = \begin{pmatrix} \varphi_1^{r,1}(k) \\ \varphi_2^{r,1}(k) \end{pmatrix}, \quad r = 0,1,2.$$

From (1) and (15) we have

$$\begin{aligned} \vec{y}^{r,1}(l) &= \sum_{n\in\mathbb{Z}} \tilde{H}^r(n-2l)\, \vec{x}(n) = \sum_{n\in\mathbb{Z}} \tilde{H}^r(n-2l) \sum_{k\in\mathbb{Z}} \tilde{\vec{\varphi}}^0(k-2n)f(k) \\ &= \sum_{k\in\mathbb{Z}} f(k) \sum_{n\in\mathbb{Z}} \tilde{H}^r(n-2l)\, \tilde{\vec{\varphi}}^0(k-2n) = \left[\tilde{\vec{\varphi}}^{r,1}(\cdot - 4l), f\right],\ r = 0,1.2, \end{aligned} \tag{16}$$

where

$$\tilde{\vec{\varphi}}^{r,1}(k) \triangleq \sum_{l\in\mathbb{Z}} \tilde{H}^r(l)\tilde{\vec{\varphi}}^0(k-2l) = \begin{pmatrix} \tilde{\varphi}_1^{r,1}(k) \\ \tilde{\varphi}_2^{r,1}(k) \end{pmatrix}, \quad r = 0,1,2.$$

Thus, we obtained six analysis and six synthesis waveforms that implement the frame expansion of a scalar signal $f(k)$ in the following way

$$\begin{aligned} f(k) &= \sum_{n\in\mathbb{Z}} \left\langle f,\ \tilde{\varphi}_1^{0,1}(\cdot-4n)\right\rangle \varphi_1^{0,1}(n-4k) + \left\langle f,\ \tilde{\varphi}_2^{0,1}(\cdot-4n)\right\rangle \varphi_2^{0,1}(n-4k) \\ &+ \left\langle f,\ \tilde{\varphi}_1^{1,1}(\cdot-4n)\right\rangle \varphi_1^{1,1}(n-4k) + \left\langle f,\ \tilde{\varphi}_2^{1,1}(\cdot-4n)\right\rangle \varphi_2^{1,1}(n-4k) \\ &+ \left\langle f,\ \tilde{\varphi}_1^{2,1}(\cdot-4n)\right\rangle \varphi_1^{2,1}(n-4k) + \left\langle f,\ \tilde{\varphi}_2^{2,1}(\cdot-4n)\right\rangle \varphi_2^{2,1}(n-4k). \end{aligned}$$

2.4 Multiscale processing

As it is common in multiwavelet theory, we extend the transform to coarser scales by recursive application of the multifilter bank to the output from the multifilter $\tilde{H}^0$. To be specific, for the second decomposition scale we have

$$\begin{aligned} \vec{y}^{r,2}(l) &= \sum_{n\in\mathbb{Z}} \tilde{H}^r(n-2l)\,\vec{y}^{0,1}(n),\ r=0,1.2, \\ \vec{x}(l) &= \sum_{r=0}^{2}\sum_{n\in\mathbb{Z}} H^0(l-2n)\sum_{m\in\mathbb{Z}} H^r(n-2m)\,\vec{y}^{r,2}(m) + \sum_{r=1}^{2}\sum_{n\in\mathbb{Z}} H^k(l-2n)\,\vec{y}^{r,1}(n). \end{aligned}$$

The analysis and synthesis waveforms of the second scale are

$$\begin{aligned} \tilde{\vec{\varphi}}^{r,2}(k) &\triangleq \sum_{l\in\mathbb{Z}} \tilde{H}^r(l)\tilde{\vec{\varphi}}^{0,1}(k-4l) = \begin{pmatrix} \tilde{\varphi}_1^{r,2}(k) \\ \tilde{\varphi}_2^{r,2}(k) \end{pmatrix}, \\ \vec{\varphi}^{r,2}(k) &\triangleq \sum_{l\in\mathbb{Z}} H^r(l)^T\vec{\varphi}^{0,1}(k-4l) = \begin{pmatrix} \varphi_1^{r,2}(k) \\ \varphi_2^{r,2}(k) \end{pmatrix}, \quad r=0,1,2. \end{aligned}$$

Then, we get the following frame expansion of the signal $f(k)$:

$$\begin{aligned} f(k) &= \sum_{n\in\mathbb{Z}}\sum_{r=0}^{2} \left\langle f,\ \tilde{\varphi}_1^{r,2}(\cdot-8n)\right\rangle \varphi_1^{r,2}(n-8k) + \left\langle f,\ \tilde{\varphi}_2^{r,2}(\cdot-8n)\right\rangle \varphi_2^{r,2}(n-8k) \\ &+ \sum_{r=1}^{2} \left\langle f,\ \tilde{\varphi}_1^{r,1}(\cdot-4n)\right\rangle \varphi_1^{r,1}(n-4k) + \left\langle f,\ \tilde{\varphi}_2^{r,1}(\cdot-4n)\right\rangle \varphi_2^{r,1}(n-4k). \end{aligned}$$

By iterating these procedures down to coarser scale J, we get the expansion

$$\begin{aligned} f(k) &= \sum_{n\in\mathbb{Z}} \left\langle f,\ \tilde{\varphi}_1^{0,J}(\cdot-2^{J+1}n)\right\rangle \varphi_1^{0,J}(\cdot-2^{J+1}n) + \left\langle f,\ \tilde{\varphi}_1^{0,J}(\cdot-2^{J+1}n)\right\rangle \varphi_1^{0,J}(\cdot-2^{J+1}n) \\ &+ \sum_{r=1}^{2}\sum_{j=1}^{J} \left\langle f,\ \tilde{\varphi}_1^{r,j}(\cdot-2^{j+1}n)\right\rangle \varphi_1^{r,j}(\cdot-2^{j+1}n) + \left\langle f,\ \tilde{\varphi}_1^{r,j}(\cdot-2^{j+1}n)\right\rangle \varphi_1^{r,j}(\cdot-2^{j+1}n), \end{aligned}$$

where

$$\begin{aligned} \tilde{\vec{\varphi}}^{r,j}(k) &\triangleq \textstyle\sum_{l\in\mathbb{Z}} \tilde{H}^r(l)\tilde{\vec{\varphi}}^{0,j-1}(k-2^jl) = \begin{pmatrix} \tilde{\varphi}_1^{r,j}(k) \\ \tilde{\varphi}_2^{r,j}(k) \end{pmatrix}, \\ \vec{\varphi}^{r,j}(k) &\triangleq \textstyle\sum_{l\in\mathbb{Z}} H^r(l)^T\vec{\varphi}^{0,j-1}(k-2^jl) = \begin{pmatrix} \varphi_1^{r,j}(k) \\ \varphi_2^{r,j}(k) \end{pmatrix}, \quad r=0,1,2,\ j=1,\ldots,J. \end{aligned} \tag{17}$$

Let $\tilde{\vec{\Phi}}^0(z) \triangleq \left(\tilde{R}^1(z), \tilde{R}^2(z)\right)^T$ and $\vec{\Phi}^0(z) \triangleq \left(R^1(z), R^2(z)\right)^T$ denote the z-transforms of the vector signals $\tilde{\vec{\varphi}}^0(k)$ and $\vec{\varphi}^0(k)$, respectively. Let $\tilde{\vec{\Phi}}^{r,j}(z)$ and $\vec{\Phi}^{r,j}(z)$ denote the z-transforms of the vector signals $\tilde{\vec{\varphi}}^{r,j}(k)$ and $\vec{\varphi}^{r,j}(k)$, respectively. Eq. (17) implies the following relations

$$\tilde{\vec{\Phi}}^{r,j}(z) = \tilde{\mathbf{H}}^{\mathbf{r}}(z^{2^j})\tilde{\vec{\Phi}}^{0,j-1}(z) = \tilde{\mathbf{H}}^{\mathbf{r}}(z^{2^j}) \prod_{n=j-1}^{1} \tilde{\mathbf{H}}^{\mathbf{0}}(z^{2^n})\tilde{\vec{\Phi}}^0(z),$$

$$\vec{\Phi}^{r,j}(z) = \mathbf{H}^{\mathbf{r}}(z^{2^j})\vec{\Phi}^{0,j-1}(z) = \mathbf{H}^{\mathbf{r}}(z^{2^j})^T \prod_{n=j-1}^{1} \mathbf{H}^{\mathbf{0}}(z^{2^n})^T\vec{\Phi}^0(z), \quad r = 0, 1, 2, \ j = 1, \ldots, J.$$

We call the signals $\tilde{\vec{\varphi}}^{r,j}(k)$ and $\vec{\varphi}^{r,j}(k)$ the discrete analysis and synthesis framelets of j-th scale, respectively.

3 Construction of multifilter banks

3.1 Filter banks with interpolatory multifilters

Definition 3.1 *We call a multifilter* $\mathbf{H}$ *interpolatory if, being applied to the upsampled subarray* $\vec{e} \triangleq \{e_1(k), e_2(k))^T\} = (x_1(2k), x_2(2k))^T, \ k \in \mathbb{Z}$, *it retains this subarray (up to multiplication by a constant diagonal matrix)*

$$Y(z) = \mathbf{H}(z)\vec{E}(z^2) = \Lambda\left(\vec{E}(z^2) + z^{-1}\vec{Y}_o(z^2)\right), \quad \Lambda \triangleq \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}.$$

It follows from (3) that the z-transform of the interpolatory multifilter $\mathbf{H}$ is

$$\mathbf{H}(z) = \Lambda\left(\mathbf{I}_2 + z^{-1}\mathbf{H}_o(z^2)\right) \text{ or } \mathbf{H}(z) = \left(\mathbf{I}_2 + z^{-1}\mathbf{H}_o(z^2)\right)\Lambda.$$

We consider perfect reconstruction multifilter banks, where the multifilters

$$\tilde{\mathbf{H}}^0(z) = \frac{\Lambda}{2}\left(\mathbf{I_2} + z^{-1}\tilde{\mathbf{A}}(z^2)\right), \quad \mathbf{H}^0(z) = \left(\mathbf{I_2} + z^{-1}\mathbf{A}(z^2)\right)\Lambda^{-1} \tag{18}$$

are interpolatory. Then, the polyphase matrices become

$$\mathbf{P}(z) = \begin{pmatrix} \mathbf{I_2}\Lambda^{-1} & \mathbf{H}_e^1(z) & \mathbf{H}_e^2(z) \\ \mathbf{A}(z)\Lambda^{-1} & \mathbf{H}_o^1(z) & \mathbf{H}_o^2(z) \end{pmatrix}, \quad \tilde{\mathbf{P}}(z) = \begin{pmatrix} \Lambda\mathbf{I_2}/2 & \Lambda\tilde{\mathbf{A}}(z)/2 \\ \tilde{\mathbf{H}}_e^1(z) & \tilde{\mathbf{H}}_o^1(z) \\ \tilde{\mathbf{H}}_e^2(z) & \tilde{\mathbf{H}}_o^2(z) \end{pmatrix}.$$

From the perfect reconstruction property (5) we have

$$\mathbf{P}_1(z)\tilde{\mathbf{P}}_1(1/z) = \frac{1}{2}\mathbf{Q}(z), \tag{19}$$

where

$$\mathbf{P}_1(z) \triangleq \begin{pmatrix} \mathbf{H}_e^1(z) & \mathbf{H}_e^2(z) \\ \mathbf{H}_o^1(z) & \mathbf{H}_o^2(z) \end{pmatrix}, \quad \tilde{\mathbf{P}}_1(z) \triangleq \begin{pmatrix} \tilde{\mathbf{H}}_e^1(z) & \tilde{\mathbf{H}}_o^1(z) \\ \tilde{\mathbf{H}}_e^2(z) & \tilde{\mathbf{H}}_o^2(z) \end{pmatrix},$$

$$\mathbf{Q}(z) \triangleq \begin{pmatrix} \mathbf{I_2} & -\tilde{\mathbf{A}}(1/z) \\ -\mathbf{A}(z) & 2\mathbf{I_2} - \mathbf{A}(z)\tilde{\mathbf{A}}(1/z) \end{pmatrix}.$$

Thus, construction of a perfect reconstruction multifilter bank with given interpolatory multifilters $\mathbf{H}^0$ and $\tilde{\mathbf{H}}^0$ is reduced to the factorization (19) of the matrix $\mathbf{Q}(z)$. We consider the block-triangle factorization:

$$\mathbf{P}_1(z) = \frac{1}{\sqrt{2}}\begin{pmatrix} \mathbf{I_2} & \mathbf{0} \\ -\mathbf{A}(z) & \mathbf{V}(z) \end{pmatrix}, \quad \tilde{\mathbf{P}}_1(z) = \frac{1}{\sqrt{2}}\begin{pmatrix} \mathbf{I_2} & -\tilde{\mathbf{A}}(z) \\ \mathbf{0} & \tilde{\mathbf{V}}(z) \end{pmatrix},$$

$$
\begin{aligned}
\tilde{\mathbf{H}}^0(z) &= \Lambda(\mathbf{I_2} + z^{-1}\tilde{\mathbf{A}}(z^2))/2, \quad \mathbf{H}^0(z) = (\mathbf{I_2} + z^{-1}\mathbf{A}(z^2))\Lambda^{-1} \\
\tilde{\mathbf{H}}^1(z) &= (\mathbf{I_2} - z^{-1}\tilde{\mathbf{A}}(z^2))/\sqrt{2}, \quad \mathbf{H}^1(z) = (\mathbf{I_2} - z^{-1}\mathbf{A}(z^2))/\sqrt{2} \\
\tilde{\mathbf{H}}^2(z) &= z^{-1}\tilde{\mathbf{V}}(z^2)/\sqrt{2}, \qquad \mathbf{H}^2(z) = z^{-1}\mathbf{V}(z^2)/\sqrt{2},
\end{aligned} \tag{20}
$$

where $\tilde{\mathbf{V}}(z)$ and $\mathbf{V}(z)$ are a pair of matrix functions that satisfy

$$\mathbf{V}(z)\tilde{\mathbf{V}}(1/z) = \mathbf{W}(z) \triangleq 2\left(\mathbf{I_2} - \mathbf{A}(z)\tilde{\mathbf{A}}(1/z)\right).$$

In this construction, the multifilters $\mathbf{H}^1$ and $\tilde{\mathbf{H}}^1$ are interpolatory.

3.2 Multifilter banks related to Hermite spline

It was described in Section 3.1 how to expand a pair of interpolatory multifilters $\tilde{\mathbf{H}}^0(z)$ and $\mathbf{H}^0(z)$ to the perfect reconstruction multifilter bank. From (20) we get that for the analysis multifilter $\tilde{\mathbf{H}}^0(z)$ to restore and the multifilter $\tilde{\mathbf{H}}^1(z)$ to eliminate some regular vector signals, the multifilter $\tilde{\mathbf{A}}(z)$ must have the following property: If the z-transform of the signal $\vec{x}(k)$ is $\vec{X}(z) = \vec{E}(z^2) + z^{-1}\vec{O}(z^2)$ then it is necessary that

$$\tilde{\mathbf{A}}(z)\vec{E}(z) \approx \vec{O}(z) \text{ and } z^{-1}\tilde{\mathbf{A}}(z)\vec{O}(z) \approx \vec{E}(z). \tag{21}$$

In [3], we derived such a multifilter from the cubic Hermite splines. We recall that the cubic Hermite spline on the equidistant grid $\{2hk\},\ k \in \mathbb{Z}$, is a continuously differentiable function, which coincides with cubic polynomials on the intervals $(2hk, 2h(k+1)),\ k \in \mathbb{Z}$. The spline $S(t),\ t \in [2hk, 2h(k+1)]$, is completely determined by its values and slopes at the boundary points $2hk$ and $2h(k+1)$.

Let $\vec{s}(k) \triangleq (s_1(k), s_(k))^T,\ k \in \mathbb{Z}$, be a vector sequence and $S(t)$ be a Hermite spline such that

$$\begin{pmatrix} S(2hk) \\ hS'(2hk) \end{pmatrix} = \vec{s}(k) \triangleq \begin{pmatrix} s_1(k) \\ s_2(k) \end{pmatrix}.$$

Given the vectors $\vec{s}(k)$ and $\vec{s}(k+1)$, we can find the values of the spline S and its derivative at the midpoints of the intervals:

$$
\begin{aligned}
S(h(2k+1)) &= \tfrac{1}{2}s_1(k) + \tfrac{1}{2}s_1(k+1) + \tfrac{1}{4}s_2(k) - \tfrac{1}{4}s_2(k+1), \\
hS'(h(2k+1)) &= -\tfrac{3}{4}s_1(k) + \tfrac{3}{4}s_1(k+1) - \tfrac{1}{4}s_2(k) - \tfrac{1}{4}s_2(k+1).
\end{aligned} \tag{22}
$$

From (22), we define the multifilter $\tilde{\mathbf{A}}(z)$ as

$$\tilde{\mathbf{A}}(z) \triangleq \frac{1}{4}\begin{pmatrix} 2(1+z) & (z-1) \\ 3(1-z) & -(1+z) \end{pmatrix}. \tag{23}$$

The multifilter $\tilde{\mathbf{A}}(z)$ has the following approximation property.

Proposition 3.1 ([3]) *Assume $Z(t)$ is a cubic polynomial and*

$$\vec{x}(k) = \begin{pmatrix} Z(hk) \\ hZ'(hk) \end{pmatrix},\ \vec{e}(k) = \vec{x}(2k),\ \vec{o}(k) = \vec{x}(2k+1),\ X(z) = E(z^2) + z^{-1}O(z^2). \tag{24}$$

Then,

$$\tilde{\boldsymbol{A}}(z)\vec{E}(z) = \vec{O}(z) \text{ and } z^{-1}\tilde{\boldsymbol{A}}(z)\vec{O}(z) = \vec{E}(z).$$

Define the multifilters $\mathbf{M}(z) \stackrel{\Delta}{=} \mathbf{I_2} + z^{-1}\tilde{\mathbf{A}}(z^2) = \sum_k z^{-k} M(k)$ and $\mathbf{L}(z) \stackrel{\Delta}{=} \mathbf{I_2} - z^{-1}\tilde{\mathbf{A}}(z^2) = \sum_k z^{-k} L(k)$. It follows from Proposition 3.1 that the multifilter $\mathbf{M}$ restores and the multifilter $\mathbf{L}$ eliminates cubic polynomials in the following sense.

Corollary 3.1 *Assume $Z(t)$ is a cubic polynomial and $\vec{x}(k)$ is given by (24). Then,*

$$\sum_k M(k-n)\vec{x}(k) = \vec{x}(n), \quad \sum_k L(k-n)\vec{x}(k) = (0,\, 0)^T.$$

The above properties of the multifilters $\mathbf{M}$ and $\mathbf{L}$ suggest to use them as the "low-pass" and the "high-pass" multifilters, respectively, provided the input vector signal $\vec{x}(k)$ has a special structure. The vector signal $\vec{x}(k)$ is derived from the original scalar signalby the pre-processing procedure. If this scalar signal represents a function from C^1 sampled on the grid $\{hk/2\}$ then the components of the vector signal $\vec{x}(k)$ have to be of the form $x_1(k) = G(hk), \quad x_2(k) \approx hG'(hk)$, where $G(t)$ is a function from C^1. In particular, if the original scalar signal is a sampled cubic polynomial then vector signal must be $\vec{x}(k) = (Z(hk), hZ'(hk))^T$, where $Z(t)$ is some cubic polynomial.

However, as a result of application of the multifilter $\mathbf{M}(1/z)$, followed by downsampling, to $\vec{x}(k) = (Z(hk), hZ'(hk))^T$, we get $\vec{m}(k) = 2(Z(2hk), hZ'(2hk))^T$. Therefore, in order to use this vector array as the input for the transform in the coarser scale we have to rescale it by multiplication with the matrix $\Lambda/2$ where

$$\Lambda \stackrel{\Delta}{=} \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \quad \Lambda^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1/2 \end{pmatrix}.$$

Let

$$\mathbf{A}(z) \stackrel{\Delta}{=} z\tilde{\mathbf{A}}(1/z) \stackrel{\Delta}{=} \frac{1}{4}\begin{pmatrix} 2(1+z) & (1-z) \\ 3(z-1) & -(1+z) \end{pmatrix}.$$

According to (18), we define the analysis and synthesis multifilters to be

$$\tilde{\mathbf{H}}^0(z) = \frac{\Lambda}{2}\left(\mathbf{I_2} + z^{-1}\tilde{\mathbf{A}}(z^2)\right) = \frac{\Lambda}{8}\begin{pmatrix} 2(z+2+1/z) & z-1/z \\ 3(1/z-z) & 4-1/z-z) \end{pmatrix},$$

$$\mathbf{H}^0(z) = \left(\mathbf{I_2} + z^{-1}\mathbf{A}(z^2)\right)\Lambda^{-1} = \begin{pmatrix} 2(z+2+1/z) & 1/z-z \\ 3(z-1/z) & 4-1/z-z) \end{pmatrix}\frac{\Lambda^{-1}}{4}, \tag{25}$$

$$\tilde{\mathbf{H}}^1(1/z) = \mathbf{H}^1(z) = \frac{1}{\sqrt{2}}\left(\mathbf{I_2} - z^{-1}\mathbf{A}(z^2)\right) = \frac{1}{4\sqrt{2}}\begin{pmatrix} 2(2-z-1/z) & z-1/z \\ 3(1/z-z) & 4+1/z+z) \end{pmatrix}, \tag{26}$$

$$\tilde{\mathbf{H}}^2(z) = z^{-1}\tilde{\mathbf{V}}(z^2)/\sqrt{2}, \qquad \mathbf{H}^2(z) = z^{-1}\mathbf{V}(z^2)/\sqrt{2}, \tag{27}$$

where $\tilde{\mathbf{V}}(z)$ and $\mathbf{V}(z)$ are derived from the factorization of the matrix function

$$\mathbf{V}(z)\tilde{\mathbf{V}}(1/z) = \mathbf{W}(z) \stackrel{\Delta}{=} 2\left(\mathbf{I_2} - \mathbf{A}(z)\tilde{\mathbf{A}}(1/z)\right) = \frac{1}{16}\begin{pmatrix} (2-z-1/z) & z-1/z \\ -3(z-1/z) & 8+2z+2/z \end{pmatrix}. \tag{28}$$

Proposition 3.2 *The multifilter $\tilde{\mathbf{H}}^0(z)$ reproduces (up to a constant factor) cubic polynomials and the multifilter $\tilde{\mathbf{H}}^1(z)$ eliminates these polynomials.*

To complete the formation of the perfect reconstruction multifilter bank, a factorization of the matrix function $\mathbf{W}(z)$ is needed. We propose two schemes for the factorization.

Let

$$\mathbf{V}(z) \stackrel{\Delta}{=} \frac{1}{4}\begin{pmatrix} v_{11}(z) & v_{12}(z) \\ v_{21}(z) & v_{22}(z) \end{pmatrix}, \quad \tilde{\mathbf{V}}(z) \stackrel{\Delta}{=} \frac{1}{4}\begin{pmatrix} \tilde{v}_{11}(z) & \tilde{v}_{12}(z) \\ \tilde{v}_{21}(z) & \tilde{v}_{22}(z) \end{pmatrix}.$$

Multifilters with the shortest impulse response: Assume $v_{12}(z) = \tilde{v}_{21}(z) \equiv 0$. Then, we get

$$v_{11}(z)\tilde{v}_{11}(1/z) = -z + 2 - 1/z, \; v_{11}(z)\tilde{v}_{12}(1/z) = z - 1/z,$$
$$v_{21}(z)\tilde{v}_{11}(1/z) = -3z + 3/z, \quad v_{22}(z)\tilde{v}_{22}(1/z) + v_{21}(z)\tilde{v}_{12}(1/z) = 8 + 2z + 2/z.$$

The simplest factorization is

$$\mathbf{V}(z) = \frac{1}{4}\begin{pmatrix} z-1 & 0 \\ 3(z+1) & z-1 \end{pmatrix}, \quad \tilde{\mathbf{V}}(z) = \frac{1}{4}\begin{pmatrix} (z-1) & (z+1) \\ 0 & z-1 \end{pmatrix}. \tag{29}$$

The impulse responses of the multifilters $\tilde{\mathbf{V}}$ and $\mathbf{V}$, defined in (29), are the shortest possible. Each comprises two terms only.

Proposition 3.3 *Let $\tilde{\mathbf{V}}(z)$ and $\mathbf{V}(z)$ be constructed according to (29). Then, the multifilter $\tilde{\mathbf{H}}^2(z) = z^{-1}\tilde{\mathbf{V}}(z^2)$ eliminates polynomials of first degree.*

Proof: Let $\vec{x}(k) = (ak + b, a)^T$. Then

$$\vec{Y}^{2,1}(z) = \tilde{\mathbf{V}}(1/z)\vec{E}(z) \Longleftrightarrow \vec{y}^{2,1}(k) = \begin{pmatrix} a(2k-1-2k-1)+2a \\ 0+a-a \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

■

A trivial factorization with maximal elimination order. We define

$$\mathbf{V}(z) \triangleq \mathbf{I}_2, \quad \tilde{\mathbf{V}}(z) \triangleq \mathbf{W}(1/z) = \frac{1}{16}\begin{pmatrix} (2 - z - 1/z) & 1/z - z \\ 3(z - 1/z) & 8 + 2z + 2/z \end{pmatrix}. \tag{30}$$

Proposition 3.4 *Let $\tilde{\mathbf{V}}(z)$ and $\mathbf{V}(z)$ be constructed according to (30). Then, the multifilter $\tilde{\mathbf{H}}^2(z) = z^{-1}\tilde{\mathbf{V}}(z^2)$ $\tilde{\mathbf{H}}^2(z)$ eliminates polynomials of third degree.*

Proof: Similar to the proof of Proposition 3.3

4 Pre-(post)processing filter banks

The goal of pre-processing is to create from the scalar signal $f(k)$ the vector signal $\vec{x}(k) \triangleq (x_1(k), x_2(k))^T, \; k \in \mathbb{Z}$, such that if $f(k)$ represents a function from C^1 sampled on the grid $\{hk/2\}$ then the components of $\vec{x}(k)$ are $x_1(k) = G(hk), \quad x_2(k) \approx hG'(hk)$, where $G(t)$ is a function from C^1. The vector signal $\vec{x}(k)$ serves as an input to the analysis multifilter bank. To retain the approximation accuracy of the multiwavelet frame transform, it is necessary for the pre-(post)processing scheme to be exact on polynomials.

Definition 4.1 *Let $f(k)$ be a polynomial of degree r sampled on the grid $\{hk/2\}$. If $x_1(k) = Z(hk)$ and $x_2(k) = hZ'(k)$, where $Z(t)$ is a polynomial of the same degree r, then we say that a pre-(post)processing scheme is exact on polynomials of degree r.*

Proposition 4.1 *If a pre-(post)processing scheme with the polyphase matrices*

$$\tilde{\mathbf{R}}(z) \triangleq \begin{pmatrix} \tilde{R}_e^1(z) & \tilde{R}_o^1(z) \\ \tilde{R}_e^2(z) & \tilde{R}_o^2(z) \end{pmatrix}, \quad \mathbf{R}(z) \triangleq \begin{pmatrix} R_e^1(z) & R_e^2(z) \\ R_o^1(z) & R_o^2(z) \end{pmatrix}$$

is exact on constants then

$$\tilde{R}_e^1(1) = \tilde{R}_o^1(1), \; R_e^1(1) = R_o^1(1) \Longleftrightarrow \sum_{l\in\mathbb{Z}} \tilde{\varphi}_1^0(2l) = \sum_{l\in\mathbb{Z}} \tilde{\varphi}_1^0(2l+1), \; \sum_{l\in\mathbb{Z}} \varphi_1^0(2l) = \sum_{l\in\mathbb{Z}} \varphi_1^0(2l+1).$$

Proof: straightforward.

We briefly outline three pre-(post)processing schemes that fit the Hermite spline- based multiwavelet frame transforms. These schemes were introduced in our previous paper [3]. One of these scheme with very short filters, is exact on quadratic polynomials and two schemes with longer filters, are exact on polynomials of fourth degree.

Haar scheme: We define the analysis and synthesis polyphase matrices as

$$\tilde{\mathbf{R}}_1(z) \triangleq \begin{pmatrix} 1 & 1 \\ -1/2 & 1/2 \end{pmatrix}, \quad \mathbf{R}_1(z) = \tilde{\mathbf{R}}_1^{-1}(1/z) = \begin{pmatrix} 1/2 & -1 \\ 1/2 & 1 \end{pmatrix}, \tag{31}$$

respectively.

Proposition 4.2 ([3]) *The pre-processing scheme (31) is exact on quadratic polynomials.*

A scheme exact on polynomials of fourth degree: The analysis and synthesis polyphase matrices are

$$\tilde{\mathbf{R}}_2(1/z) \triangleq \frac{1}{96}\begin{pmatrix} 1/z+24-z & z+24-1/z \\ -96 & 96 \end{pmatrix}, \tag{32}$$
$$\mathbf{R}_2(z) = \tilde{\mathbf{R}}_2^{-1}(1/z) = \frac{1}{48}\begin{pmatrix} 96 & z-24-1/z \\ 96 & z+24-1/z \end{pmatrix},$$

respectively.

Proposition 4.3 ([3]) *The pre-processing scheme (32) is exact on polynomials of fourth degree.*

Another scheme that is exact on polynomials of fourth degree: The analysis and synthesis polyphase matrices are

$$\tilde{\mathbf{R}}_3(z) \triangleq \frac{1}{64}\begin{pmatrix} 18 & 18 \\ 1/z-64-z & 1/z+64-z \end{pmatrix}, \tag{33}$$
$$\mathbf{R}_3(z) = \tilde{\mathbf{R}}_3^{-1}(1/z) = \frac{1}{36}\begin{pmatrix} z+64-1/z & -18 \\ 1/z+64-z & 18 \end{pmatrix},$$

respectively.

Proposition 4.4 ([3]) *The pre-processing scheme (33) is exact on polynomials of fourth degree.*

5 Properties of analysis and synthesis framelets

In this section we show that the waveforms generated by the designed multifilter banks possess valuable properties for signal processing: they have linear phase and short support. Analysis framelets have four vanishing moments, and the synthesis framelets converge to continuous functions that are Hermite splines.

Proposition 5.1 *All the discrete framelets $\tilde{\varphi}_1^{r,j}(k)$, $\varphi_1^{r,j}(k)$, $r = 0,1,2,\ j \in \mathbb{N}$, generated by the multifilter banks in Sections 3 and 4 are symmetric, whereas the framelets $\tilde{\varphi}_2^{r,j}(k)$, $\varphi_2^{r,j}(k)$ are antisymmetric.*

Proof: Assume that the Haar pre-processing scheme is applied. It follows from (31) that the z-transform of the pre-processing filter bank is

$$\begin{pmatrix} \tilde{R}^1(z) \\ \tilde{R}^2(z) \end{pmatrix} \begin{pmatrix} z^{-1}+1 \\ z^{-1}/2-1/2 \end{pmatrix} = \begin{pmatrix} \tilde{\Phi}_1^0(z) \\ \tilde{\Phi}_2^0(z) \end{pmatrix},$$

where $\tilde{\Phi}_1^0(z)$ and $\tilde{\Phi}_2^0(z)$ are the z-transforms of the pre-processing framelets $\tilde{\varphi}_1^0(k))$ and $\tilde{\varphi}_2^0(k)$, respectively. The function $\tilde{\Phi}_1^0(z)$ is symmetric and $\tilde{\Phi}_2^0(z)$ is antisymmetric about the inversion $z \to 1/z$ (up to a shift). The z-transform of the framelet $\tilde{\varphi}^{0,1}$

$$\begin{pmatrix} \tilde{\Phi}_1^{0,1}(z) \\ \tilde{\Phi}_2^{0,1}(z) \end{pmatrix} = \tilde{\mathbf{H}}^{\mathbf{0}}(z^2) \begin{pmatrix} \tilde{\Phi}_1^0(z) \\ \tilde{\Phi}_2^0(z) \end{pmatrix} = \frac{\Lambda}{8} \begin{pmatrix} 2(z^2+2+1/z^2)\Phi_1^0(z) + (z^2-1/z^2)\tilde{\Phi}_2^0(z) \\ 3(1/z^2-z^2)\tilde{\Phi}_1^0(z) + (4-1/z^2-z^2)\tilde{\Phi}_2^0(z) \end{pmatrix}$$
$$= \frac{1}{16} \begin{pmatrix} 3z^2+5z+8+8z^{-1}+5z^{-2}+3z^{-3} \\ 5z^2+7z+4-4z^{-1}-7z^{-2}-5z^{-3} \end{pmatrix}.$$

Hence, $\tilde{\Phi}_1^{0,1}(z)$ is symmetric and $\tilde{\Phi}_2^{0,1}(z)$ is antisymmetric about the inversion $z \to 1/z$. The same can be said about $\tilde{\Phi}_1^{0,j}(z)$ and $\tilde{\Phi}_2^{0,j}(z)$ due to the relation $\tilde{\vec{\Phi}}^{0,j}(z) = \tilde{\mathbf{H}}^{\mathbf{r}}(z^{2^j})\tilde{\vec{\Phi}}^{0,j-1}(z)$. Thus, framelets $\tilde{\varphi}_1^{0,j}(k),\ j \in \mathbb{Z}$, are symmetric and $\tilde{\varphi}_2^{0,j}(k)$ are antisymmetric about $1/2$. The proof of this property for the rest of the analysis framelets, for the synthesis framelets and the fourth-order pre(post)-processing schemes is similar.■

5.1 Vanishing moments of discrete analysis framelets

Definition 5.1 *We say that a signal $q = \{g(k)\}_{k\in\mathbb{Z}}$ has r vanishing moments if for any polynomial $P_r(t)$ of degree not exceeding $r-1$ $\sum_{k\in\mathbb{Z}} g(k)P_r(hk) = 0$.*

Proposition 5.2 *Assume that Haar pre-processing scheme (31) is used. Then, the discrete framelets $\varphi_m^{1,j}$, $m = 1,2,\ j \in \mathbb{N}$, have three vanishing moments. The same is true for the discrete framelets $\varphi_m^{2,j}$ provided that $\tilde{\mathbf{H}}^2(z) = z^{-1}\tilde{\mathbf{V}}(z^2)/\sqrt{2}$ and $\tilde{\mathbf{V}}$, defined in (30), is used. If $\tilde{\mathbf{V}}$, defined in (29), is used then the discrete framelets $\varphi_m^{2,j}$ have two vanishing moments.*

Proof: Let the signal $f(k) \stackrel{\Delta}{=} P_2(hk/2)$ be a sampled quadratic polynomial and $\vec{x}(k) = (x_1(k), x_2(k))^T$, $k \in \mathbb{Z}$, be the output from the pre-processing filter bank. Recall that $x_1(k) = Z_2(hk)$ is a sampled quadratic polynomial and $x_2(k) = hZ_2'(hk)$. Proposition 3.2 implies that

$$\vec{y}^{0,1}(k) \stackrel{\Delta}{=} \sum_{n\in\mathbb{Z}} \tilde{H}^0(n-2k)\,\vec{x}(n) = \begin{pmatrix} Z_2(2hk) \\ 2hZ_2'(2hk) \end{pmatrix},$$
$$\vec{y}^{0,j-1}(k) \stackrel{\Delta}{=} \sum_{n\in\mathbb{Z}} \tilde{H}^0(n-2k)\,\vec{y}^{0,j-1}(n) = \begin{pmatrix} Z_2(nh_{j-1}) \\ h2^{j-1}Z_2'(nh2^{j-1}) \end{pmatrix}.$$

For $j = 1$ we have from (16)

$$\begin{pmatrix} \langle\varphi_1^{1,1}, f\rangle \\ \langle\varphi_2^{1,1}, f\rangle \end{pmatrix} = \sum_{n\in\mathbb{Z}} \tilde{H}^1(n) \begin{pmatrix} Z_2(nh2) \\ h2Z_2'(nh2) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

because the multifilter $\tilde{\mathbf{H}}^1$ eliminates polynomials up to third degree. For $j > 1$ we have

$$\begin{pmatrix} \langle\varphi_1^{1,j}, f\rangle \\ \langle\varphi_2^{1,j}, f\rangle \end{pmatrix} = \sum_{n\in\mathbb{Z}} \tilde{H}^1(n)\vec{y}^{0,j-1}(n) = \sum_{n\in\mathbb{Z}} \tilde{H}^1(n) \begin{pmatrix} Z_2(nh2^{j-1}) \\ h2^{j-1}Z_2'(nh2^{j-1}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

If the multifilter $\tilde{\mathbf{H}}^2$ is designed according to (30) then it eliminates polynomials up to third degree. From the same calculations as above we get

$$\begin{pmatrix} \langle \varphi_1^{2,j}, f\rangle \\ \langle \varphi_2^{2,j}, f\rangle \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

However, if $\tilde{\mathbf{H}}^2$ is designed according to (29) then it eliminates only polynomials of first degree. Therefore, in this case, the discrete framelets $\varphi_m^{2,j}$, $m = 1, 2$ have two vanishing moments.■

Proposition 5.3 *Assume that either the pre-processing scheme (32) or the scheme (33) is used. Then, the discrete framelets $\varphi_m^{1,j}$, $m = 1, 2$, $j \in \mathbb{N}$, have four vanishing moments. The same is true for the discrete framelets $\varphi_m^{2,j}$ provided that $\tilde{\mathbf{H}}^2(z) = z^{-1}\tilde{\mathbf{V}}(z^2)/\sqrt{2}$ and $\tilde{\mathbf{V}}$, defined in (30), is used. If $\tilde{\mathbf{V}}$, defined in (29), is used then the discrete framelets $\varphi_m^{2,j}$ have two vanishing moments.*

Proof: Similar to the proof of Proposition 5.2.

5.2 Scaling function and framelets related to the synthesis filter bank

A classical method to derive a scaling function from a low-pass filter consists of recursive application of the filter to the impulse signal and substitution of the values of the output signals into the diadic rational points of the real line [11]. The wavelets are obtained by the application of the high-pass filters to the derived scaling function. A similar scheme works for the multifilter case. We present the construction for the synthesis filter banks that results in Hermite splines, which allow explicit representation.

Let $\vec{x}^0 \triangleq \left\{ \left(x_1^0(k), x_2^0(k)\right)^T \right\}_{k\in\mathbb{Z}}$ be a vector array and $S(t)$ be a Hermite cubic spline interpolating the array $\vec{x}^0$, that is

$$\left\{ \vec{S}(k) = \vec{x}^0(k) \right\}_{k\in\mathbb{Z}} \text{ where } \vec{S}(t) \triangleq \begin{pmatrix} S(t) \\ S'(t) \end{pmatrix}.$$

Application of the synthesis multifilters $\mathbf{H}^r$, $r = 0, 1, 2$, to the upsampled vector signal $\vec{x}^0$ produces the vector arrays $\vec{x}^{r,1}(k) = \sum_{n\in\mathbb{Z}} H^r(k-2n)\vec{x}^0(n)$. We define the following sequences of the vector arrays

$$\vec{x}^{r,j}(k) = \sum_{n\in\mathbb{Z}} H^0(k-2n)\vec{x}^{r,j-1}(n), \; j = 2, 3, ..., \; r = 0, 1, 2.$$

Proposition 5.4 *If the synthesis multifilters $\mathbf{H}^r$ are defined by Eq. (25) then*

$$\vec{x}^{r,j}(k) = \begin{pmatrix} S^r\left(k2^{-j}\right) \\ 2^{-j}S^{r\prime}\left(k2^{-j}\right) \end{pmatrix}, \quad j = 2, 3, ..., \; r = 0, 1, 2, \tag{34}$$

where $S^r(t)$ is a Hermite cubic spline such that

$$\begin{pmatrix} S^r(k/2) \\ S^{r\prime}(k/2)/2 \end{pmatrix} = \vec{x}^{r,1}(k).$$

Proof: We begin with the case $j=2$. Due to Eqs. (25), (22) and Proposition 3.1

$$\vec{x}^{r,2}(2k) = \Lambda^{-1}\vec{x}^{r,1}(k) = \begin{pmatrix} S^r(2kh) \\ hS^{r\prime}(2kh) \end{pmatrix},\ h=2^{-2},\ r=0,1,2\ k\in\mathbb{Z},$$

$$\vec{x}^{r,2}(2k+1) = \mathbf{A}\cdot\Lambda^{-1}\vec{x}^{r,1}(k) = \mathbf{A}\cdot\begin{pmatrix} S^r(2kh) \\ hS^{r\prime}(2kh) \end{pmatrix} = \begin{pmatrix} S^r\left((2k+1)2^{-2}\right) \\ 2^{-2}S'\left((2k+1)2^{-2}\right) \end{pmatrix}.$$

Assume that (34) is true for $j=J-1$. Repeating the above calculations with $h=2^{-J}$ instead of $h=2^{-2}$, we obtain (34) for $j=J$.■

Remark. Obviously, the spline $S^0(t)$ coincides with the spline $S(t)$ that interpolates the array $\vec{x}^0$.

Denote

$$\vec{\xi}_1^0 \triangleq \left\{\begin{pmatrix} \delta(k) \\ 0 \end{pmatrix}\right\}_{k\in\mathbb{Z}},\quad \vec{\xi}_2^0 \triangleq \left\{\begin{pmatrix} 0 \\ \delta(k) \end{pmatrix}\right\}_{k\in\mathbb{Z}},$$

where $\delta(k)$ is the Kronecker delta. Let $\sigma_1^0(t)$ and $\sigma_2^0(t)$ be Hermite splines, which interpolate $\vec{\xi}_1$ and $\vec{\xi}_2$, respectively. These splines are supported on the interval $(-1,1)$ and have explicit representations

$$\sigma_1^0(t) = \begin{cases} (t+1)^2(-2t+1), & t\in[-1,0], \\ (t-1)^2(2t+1), & t\in[0,1], \end{cases} \qquad \sigma_1^0(t) = \begin{cases} (t+1)^2 t, & t\in[-1,0], \\ (t-1)^2 t, & t\in[0,1]. \end{cases} \tag{35}$$

Denote by $\sigma_m^r(t)$ the Hermite cubic spline such that

$$\begin{pmatrix} \sigma_m^r(k/2) \\ (\sigma_m^r)'(k/2)/2 \end{pmatrix} = \vec{\xi}_m^{r,1}(k) \triangleq \sum_{n\in\mathbb{Z}} H^r(k-2n)\vec{\xi}_m^0(n),\ m=1,2,\ r=0,1,2. \tag{36}$$

Let $R^m(z),\ m=1,2$, be the transfer functions of a FIR post-processing filter bank and $\vec{\Phi}^0(z) = \left(R^1(z),R^2(z)\right)^T$ be the z-transform of the vector signal $\vec{\varphi}^0(k)$. Let $\vec{\varphi}^{r,j}(k) = \left(\varphi_1^{r,j}(k),\varphi_2^{r,j}(k)\right)^T$, $r=0,1,2$, be the discrete synthesis framelets of the j-th scale, as defined in (17). Let $\rho_m^{r,j}(t),\ j\in\mathbb{Z}_+,\ m=1,2$, be continuous piecewise linear functions such that $\rho_m^{r,j}(k2^{-j-1}) = \varphi_m^{r,j}(k),\ k\in\mathbb{Z},\ m=1,2$.

Theorem 5.1 *If the transfer function* $R^1(1)=c$ *then the sequences* $\rho_m^{r,j}(t),\ j\in\mathbb{Z}_+,\ m=1,2$, *converge uniformly and*

$$\lim_{j\to\infty}\rho_m^{r,j}(t) = \frac{c}{2}\sigma_m^r(t). \tag{37}$$

Remark It is worth to note that the limit functions for $\rho_m^{r,j}(t)$ does not depend on the employed post-processing scheme. Choice of the post-processing scheme affects only the speed of convergence.

Proof: in Appendix.

The Hermite spline $S(t)$ such that

$$\begin{pmatrix} S(k/2) \\ S'(k/2)/2 \end{pmatrix} = \vec{\eta}(k) = \begin{pmatrix} \eta_1(k) \\ \eta_2(k) \end{pmatrix},\quad k\in\mathbb{Z}$$

can be expanded into the sum

$$S(t) = \sum_{k\in\mathbb{Z}} \eta_1(k)\sigma_1^0(2t-k) + \eta_2(k)\sigma_2^0(2t-k).$$

It is readily seen from Eq. (36) that

$$\vec{\xi}_1^{r,1}(k) = \begin{pmatrix} h_{11}^r(k) \\ h_{21}^r(k) \end{pmatrix}, \quad \vec{\xi}_2^{r,1}(k) = \begin{pmatrix} h_{12}^r(k) \\ h_{22}^r(k) \end{pmatrix}, \quad k \in \mathbb{Z}, \;\; r = 0, 1, 2.$$

Hence, it follows that

$$\begin{array}{l} \sigma_1^r(t) = \sum_{k\in\mathbb{Z}} h_{11}^r(k)\sigma_1^0(2t-k) + h_{21}^r(k)\sigma_2^0(2t-k), \\ \sigma_2^r(t) = \sum_{k\in\mathbb{Z}} h_{12}^r(k)\sigma_1^0(2t-k) + h_{22}^r(k)\sigma_2^0(2t-k). \end{array} \tag{38}$$

Define the vector functions

$$\vec{\varphi}(t) = \begin{pmatrix} \varphi_{1(t)} \\ \varphi_2(t) \end{pmatrix} \triangleq \begin{pmatrix} \sigma_1^0(t) \\ \sigma_2^0(t) \end{pmatrix}, \quad \vec{\psi}^r(t) = \begin{pmatrix} \psi_2(t) \\ \psi_2(t) \end{pmatrix} \triangleq \begin{pmatrix} \sigma_1^r(t) \\ \sigma_2^r(t) \end{pmatrix}, \quad r = 1, 2. \tag{39}$$

Equation (38) implies the following refinement relations

$$\vec{\varphi}(t) = \sum_{k\in\mathbb{Z}} H^0(k)^T \vec{\varphi}(2t-k), \quad \vec{\psi}^r(t) = \sum_{k\in\mathbb{Z}} H^r(k)^T \vec{\varphi}(2t-k), \; r = 1, 2.$$

Thus, $\vec{\varphi}(t)$ is a scaling vector function and $\vec{\psi}^r(t)$, $r = 1, 2$, are continuous vector framelets.

6 Examples

Scaling vector function, generated by the synthesis multifilter $\mathbf{H}^0$ (Eq. (25)), consists of the pair of Hermite cubics $\sigma_1^0(t)$, $\sigma_2^0(t)$ (Eq. (35)), which are supported on the interval [-1,1] and belong to the space R^1. They are displayed in Figure 3

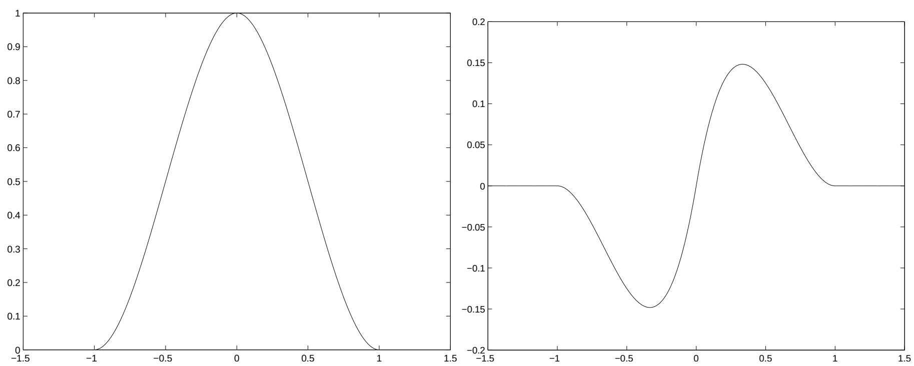

Figure 3: Pair of scaling functions generated by the synthesis multifilter bank. Left: $\varphi_1(t) = \sigma_1^0(t)$, right: $\varphi_2(t) = \sigma_2^0(t)$.

Framelet $\vec{\psi}^1$ (Eq. (39)), generated by the synthesis multifilter, $\mathbf{H}^1$ (Eq. (26)) consists of the pair of Hermite splines $\psi_1^1(t)$, $\psi_2^1(t)$, which are supported on the interval [-1,1] and belong to the space C^1. Equations (26) and (38) imply the following representation of the framelet:

$$\begin{aligned} \psi_1^1(t) &= \frac{1}{4\sqrt{2}} \left(-2\sigma_1^0(2t-1) + 4\sigma_1^0(2t) - 2\sigma_1^0(2t-1) + 3\sigma_2^0(2t-1) - 3\sigma_2^0(2t+1)\right), \\ \psi_2^1(t) &= \frac{1}{4\sqrt{2}} \left(\sigma_1^0(2t+1) - \sigma_1^0(2t-1) + \sigma_2^0(2t+1) + 4\sigma_2^0(2t) + \sigma_2^0(2t-1)\right). \end{aligned}$$

They are displayed in Figure 4.

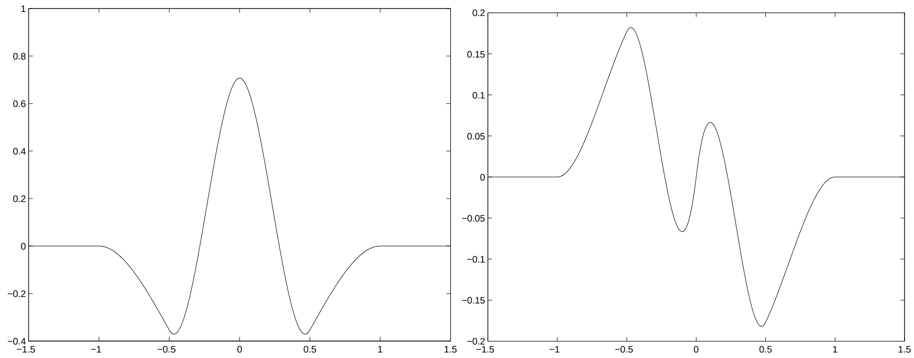

Figure 4: Pair of framelets generated by the synthesis multifilter $\mathbf{H}^1$. Left: $\psi_1^1(t)$, right: $\psi_2^1(t)$.

Framelets $\vec{\psi}^2$, generated by the synthesis multifilter $\mathbf{H}^2$ (Eq. (39)), consists of the pair of Hermite splines $\psi_1^2(t)$, $\psi_2^2(t)$. We presented in Section 3 two modes (Eqs. (29) and (30)) of the factorization of the matrix $\mathbf{W}(z)$ (Eq. (28)), which lead to two different framelets $\vec{\psi}^2$.

Framelet resulting from factorization (29):

$$\psi_1^2(t) = \frac{1}{4\sqrt{2}}\left(\sigma_1^0(2t+1) - \sigma_1^0(2t-1) + 3\sigma_2^0(2t-1) + 3\sigma_2^0(2t+1)\right),$$
$$\psi_2^2(t) = \frac{1}{4\sqrt{2}}\left(\sigma_2^0(2t+1) - \sigma_2^0(2t-1)\right).$$

The framelets are supported on the interval [-1,1]. They are displayed in Figure 5.

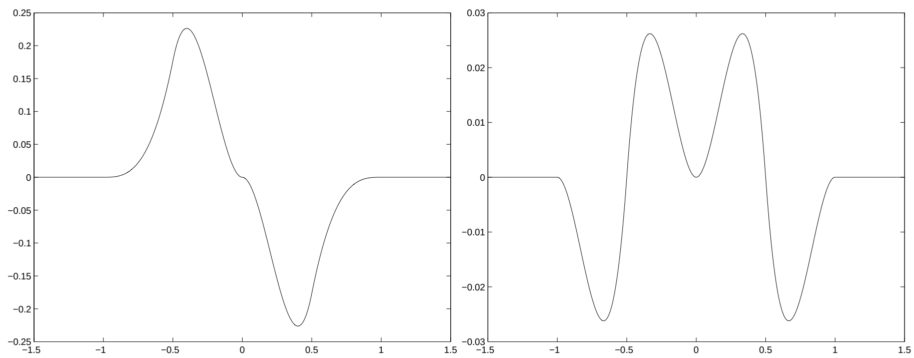

Figure 5: Pair of framelets generated by the synthesis multifilter $\mathbf{H}^2$ using factorization (29). Left: $\psi_1^2(t)$, right: $\psi_2^2(t)$.

Framelet resulting from factorization (30):

$$\psi_1^2(t) = \frac{1}{\sqrt{2}}\sigma_1^0(2t-1), \quad \psi_2^2(t) = \frac{1}{\sqrt{2}}\sigma_2^0(2t-1).$$

The framelets are supported on the interval [0,1]. They are displayed in Figure 6.

Conclusions

In this paper we presented a new group of multiwavelet frames in the space of square summable discrete signals. The paper can be regarded as an extension of our previous paper [3] that was devoted

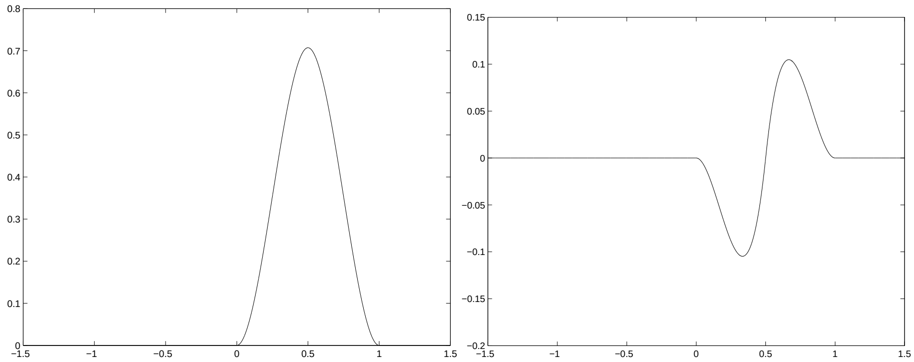

Figure 6: Pair of framelets generated by the synthesis multifilter $\mathbf{H}^2$ using factorization (30). Left: $\psi_1^2(t)$, right: $\psi_2^2(t)$.

to the design of multiwavelet bases in the signal space using cubic Hermite splines. The redundancy inherent in frame expansions of signals provides more flexibility in design and implementation of the transforms compared to the one-to-one bases expansions. The frame multifilters, introduced in Section 3, are shorter and have simpler structure than the basis multifilters in [3]. The waveforms have shorter supports. On the other hand, diversity of the involved waveforms enhances the adaptation abilities of the expansion. In addition, missed coefficients can be restored. Therefore, the multiwavelet frames have a potential to be useful in signal denoising, feature extraction and error correction while the signal is transmitted through a noisy channel. We conducted successful experiments on erasure recovery [5] using the wavelet frames designed in [4]. The outputs from the experiments with multiwavelet frames will be presented in our next paper.

The method to construct multiwavelet frames starting from the interpolatory multifilters, which is developed in the paper, is generic. It can be performed using interpolating aggregates other than cubic Hermite splines. On the other hand, employing Hermite splines of higher order will lead to better approximation accuracy and wider diversity of waveforms.

Appendix

Proof of Theorem 5.1: Define for $j = 2, 3, ...$ the sequences of vector arrays

$$\vec{\xi}_m^{r,j}(k) \triangleq \sum_{n\in\mathbb{Z}} H^0(k-2n)\vec{\xi}_m^{r,j-1}(n) \Longleftrightarrow \vec{\Xi}_m^{r,j}(z) = \prod_{l=0}^{j-2} \mathbf{H}^0(z^{2^l})\mathbf{H}^r(z^{2^{j-1}})\vec{\Xi}_m^0(z^{2^j}), \quad m = 1, 2, \; r = 0, 1, 2.$$

Due to Proposition 5.4,

$$\vec{\xi}_m^{r,j}(k) = \begin{pmatrix} \sigma_m^r(k2^{-j}) \\ 2^{-j}(\sigma_m^r)'(k2^{-j}) \end{pmatrix}.$$

Assume that the impulse response $\varphi_l^0(k), \; l = 1, 2$, of the FIR filters $R^l(z)$ vanish when $|k| > K$. Application of the post-processing filter bank to the vector arrays $\vec{\xi}_m^{rj}(k)$ produces the scalar arrays

$$f_m^{r,j}(k) = \sum_{|2l-k|\le K} \sigma_m^r(l2^{-j})\varphi_1^0(k-2l) + 2^{-j} \sum_{|2l-k|\le K} (\sigma_m^r)'(l2^{-j})\varphi_2^0(k-2l). \tag{40}$$

On the other hand, Eq. (13) implies

$$F_m^{r,j}(z) = \langle \vec{\Xi}_m^{r,j}(z^2), \Phi^0(z)\rangle_2 = \langle \vec{\Xi}_m^0(z^{2^{j+1}}), \mathbf{H}^r(z^{2^j})^T \prod_{l=j-1}^{1} \mathbf{H}^0(z^{2^l})^T \vec{\Phi}^0(z)\rangle_2 = \langle \vec{\Xi}_m^0(z^{2^{j+1}}), \vec{\Phi}^{r,j}(z)\rangle_2.$$

Hence, it follows that

$$f_1^{r,j}(k) = \varphi_1^{r,j}(k), \quad f_2^{r,j}(k) = \varphi_2^{r,j}(k).$$

Proposition 4.1 implies that $\sum_{l\in\mathbb{Z}} \varphi_1^0(2l) = \sum_{l\in\mathbb{Z}} \varphi_1^0(2l+1) = c/2$. Let M be a positive constant such that $|(\sigma_m^r)'(t))| < M,\ t \in \mathbb{R},\ m = 1, 2,\ r = 0, 1, 2.$ Let L be a positive constant such that $\sum_{l\in\mathbb{Z}} |\varphi_2^0(l)| < L$. Then, we derive from (40):

$$f_m^{r,j}(k) = \sigma_m^r(k2^{-j-1}) \sum_{|2l-k|\le K} \varphi_1^0(k - 2l) + 2^{-j}\Upsilon(k) = \frac{c}{2}\sigma_m^r(k2^{-j-1}) + 2^{-j}\Upsilon(k),$$

where $|\Upsilon(k)| < (K + L)M$. Hence, Eq. (37) follows.■

References

[1] B. K. Alpert, *A class of bases in L^2 for the sparse representation of integral operators*, SIAM Journal on Mathematical Analysis **24**, (1993), 246-262.

[2] B. Alpert, G. Beylkin, R. Coifman, and V. Rokhlin, *Wavelet-like bases for the fast solution of second-kind integral equations*, SIAM Journal on Scientific Computing 14, 159-184 (1993).

[3] A. Z. Averbuch, V. A. Zheludev, *Lifting scheme for biorthogonal multiwavelets originated from Hermite splines*, IEEE Trans. Sign. Proc. **50**, No.3, March 2002, 487-500.

[4] A. Z. Averbuch, V. A. Zheludev, , T. Cohen *Interpolatory frames in signal space* to appear in IEEE Trans. Sign. Proc..

[5] O. Amrani2, A. Z. Averbuch, T. Cohen and V. A. Zheludev, *Symmetric interpolatory framelets and their erasure correction properties*, submitted to IEEE Trans. Sign. Proc..

[6] H. Bölskei, F. Hlawatsch, H. Feichtinger, *Frame-theoretic analysis of oversampled filter banks,* IEEE Trans. Sign. Proc., **46**, No.12,(1998), 3256-3268.

[7] C. K. Chui and W. He, *Compactly supported tight frames associated with refinable functions,* Applied and Comp. Harmonic Analysis, **8** (2000), 293-319.

[8] C. K. Chui, W. He, and J. Stöckler, *Compactly supported tight and sibling frames with maximum vanishing moments,* Applied and Comp. Harmonic Analysis, **13** (2002), 224-262.

[9] M. Cotronei, D. Lazzaro, L. B. Montefusco and L. Puccio, *Image compression through embedded multiwavelet transform coding*, IEEE Transactions on Image Processing, **9**, (2000), 184 - 189.

[10] Z. Cvetković and M. Vetterli, *Oversampled filter banks*, IEEE Trans. Sign. Proc., **46**, No.5,(1998), 1245-1255.

[11] I. Daubechies, Ten lectures on wavelets, SIAM. Philadelphia, PA, 1992.

[12] I. Daubechies and Bin Han, A. Ron, and Z. Shen, *Framelets: MRA-based constructions of wavelet frames,* Applied and Computational Harmonic Analysis, **14** (2003), 1–46.

[13] I. Daubechies and Bin Han, *Pairs of dual wavelet frames from any two refinable functions,* Constructive Approximation, **20**, (2004), 325–352.

[14] B. Han and Q. Mo, *Multiwavelet Frames from Refinable Function Vectors*, Advances in Computational Mathematics, **18**, (2003), 211-245.

[15] M. B. Martin, A. E. Bell, *New image compression techniques using multiwavelets and multiwavelet packets*, IEEE Transactions on Image Processing, **10**, (2001), 500 - 510.

[16] J. S. Geronimo, D. P. Hardin and P. R. Massopust, *Fractal functions and wavelet expansions based on several scaling functions* J. Approx. Theory, **78**, (1994), 373-401.

[17] S. S. Goh, Q. Jiang, T. Xia, *Construction of biorthogonal multiwavelets using the lifting scheme*, Appl. Comput. Harm. Anal. **9**, (2000), 336-352.

[18] T. N. T. Goodman and S. L. Lee, *Wavelets of multiplicity r* , Trans. Amer. Math. Soc., **342**, 1994, 307-324.

[19] B. Han, Q. Jiang, *Multiwavelets on the interval,* Applied and Computational Harmonic Analysis, **12**, (2002), 100-127

[20] L. Hervé, *Multi-resolution analysis of multiplicity d: Application to dyadic interpolation*, Appl. Comput. Harm. Anal. **1**, (1994), 299-315.

[21] A. Ron and Z. Shen, *Affine systems in $L^2(\mathbb{R})$ II: dual system,* J. Fourier Anal. and Appl., **3**, (1997), 617–637.

[22] A. P. Petukhov, *Explicit construction of framelets*, Applied and Comp. Harmonic Analysis, **11** (2001), 313-327.

[23] A. P. Petukhov, *Symmetric framelets*, Constr. Approx., **19**, (2003), 309-328.

[24] A. Ron and Z. Shen, Compactly supported tight affine spline frames in$L^2(\mathbb{R})$, Mathematics of Computation, **67** (1998), 191-207.

[25] G. Strang and V. Strela, *Finite element multiwavelets*, in Approximation Theory, Wavelets and Applications, S. P. Singh (ed.), Kluwer Academic Publ. (1995) 485-496.

[26] W. Dahmen, B. Han, R.-Q. Jia, A. Kunoth, *Biorthogonal multiwavelets on the interval: cubic Hermite splines.* Constr. Approx. **16**, (2000), 221–259.