Wavelet Bi-frames with Uniform Symmetry for Curve Multiresolution Processing

Qingtang Jiang *

Abstract

This paper is about the construction of univariate wavelet bi-frames with each framelet being symmetric. As bivariate filter banks are used for surface multiresolution processing, it is required that the corresponding decomposition and reconstruction algorithms have high symmetry so that it is possible to design the corresponding multiresolution algorithms for extraordinary vertices. For open surfaces, special multiresolution algorithms are designed to process boundary vertices. When the multiresolution algorithms derived from univariate wavelet bi-frames are used as the boundary algorithms, it is desired that not only the scaling functions but also all framelets be symmetric. In addition, the algorithms for curve/surface multiresolution processing should be given by templates so that they can be easily implemented.

In this paper, first, by appropriately associating the lowpass and highpass outputs to the nodes of $\mathbf{Z}$, we show that both biorthogonal wavelet multiresolution algorithms and bi-frame multiresolution algorithms can be represented by templates. Then, using the idea of the lifting scheme, we provide frame algorithms given by several iterative steps with each step represented by a symmetric template. Finally, with the given templates of algorithms, we obtain the corresponding filter banks and construct bi-frames based on their smoothness and vanishing moments. Two types of symmetric bi-frames are studied in this paper. In order to provide a clearer picture on the template-based procedure for bi-frame construction, in this paper we also consider the template-based construction of biorthogonal wavelets. The approach of the template-based bi-frame construction introduced in this paper can be extended easily to the construction of bivariate bi-frames with high symmetry for surface multiresolution processing.

Key words and phrases: biorthogonal wavelets, wavelet bi-frames, affine bi-frame, dual wavelet frames, 4-point interpolatory scheme-based bi-frames, multiresolution algorithm templates, lifting scheme, curve multiresolution processing, surface multiresolution processing.

AMS 2000 Math Subject Classification: 42C40, 65T60, 68U07, 65D17

1 Introduction

This paper studies the biorthogonal wavelet (affine) frames for curve multiresolution processing. Compared with (bi)orthogonal wavelet systems, the elements in a frame system may be linearly dependent, namely, frames can be redundant. The redundancy property is not only useful in some applications (see e.g., [4]-[7], [53]), it also provides a flexibility for the construction of framelets with short support. The property of short support, or equivalently the small size of templates of frame multiresolution algorithms, is important in curve/surface multiresolution processing.

*Q. Jiang is with the Department of Mathematics and Computer Science, University of Missouri–St. Louis, St. Louis, MO 63121, USA, e-mail: jiangq@umsl.edu.

Let $\langle \cdot, \cdot \rangle$ and $\| \cdot \|_2 := \langle \cdot, \cdot \rangle^{\frac{1}{2}}$ denote the inner product and the norm of $L^2(\mathbb{R})$. A system $G \subset L^2(\mathbb{R})$ is called a frame of $L^2(\mathbb{R})$ if there are two positive constants A and B such that

$$A\|f\|_2^2 \le \sum_{g \in G} |\langle f, g\rangle|^2 \le B\|f\|_2^2, \quad \forall f \in L^2(\mathbb{R}).$$

When $A = B$, G is called a tight frame. The reader is referred to [1], [9], [16], [19], [26], [35], [47], [48] for discussions on frames. In this paper, we consider wavelet (or affine) frames that are generated by the dilations and shifts of a set of functions. More precisely, for a function f on $\mathbb{R}$, denote $f_{j,k}(x) = 2^{j/2} f(2^j x - k)$. Functions $\psi^{(1)}, \psi^{(2)}$ on $\mathbb{R}$ are called wavelet framelets (or generators), just called framelets in this paper, if $G = \{\psi^{(1)}_{j,k}(x), \psi^{(2)}_{j,k}(x)\}_{j,k \in \mathbf{Z}}$ is a frame. In this case, G is called a wavelet (or an affine) frame. A wavelet frame could be generated by more than two framelets. In this paper we focus on frames with two framelets. There are many papers on the theory and construction of wavelet frames, see e.g., [3], [8], [10]-[16], [20]-[22], [24], [27]-[34], [39], [40], [43], [46]-[52].

For a sequence $\{p_k\}_{k\in\mathbf{Z}}$ of real numbers with finitely many p_k nonzero, let $p(\omega)$ denote the finite impulse response (FIR) filter (also called symbol) with its impulse response coefficients p_k (here a factor 1/2 is multiplied for convenience):

$$p(\omega) = \frac{1}{2} \sum_{k \in \mathbf{Z}} p_k e^{-ik\omega}.$$

For an FIR filter bank $\{p(\omega), q^{(1)}(\omega), q^{(2)}(\omega)\}$, called a **frame filter bank** in this paper, denote

$$M_{p,q^{(1)},q^{(2)}}(\omega) = \begin{bmatrix} p(\omega) & p(\omega+\pi) \\ q^{(1)}(\omega) & q^{(1)}(\omega+\pi) \\ q^{(2)}(\omega) & q^{(2)}(\omega+\pi) \end{bmatrix}. \tag{1}$$

A pair of frame filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ is said to be **biorthogonal** if $M_{p,q^{(1)},q^{(2)}}(\omega)$ and $M_{\widetilde{p},\widetilde{q}^{(1)},\widetilde{q}^{(2)}}(\omega)$ defined by (1) satisfy

$$M_{p,q^{(1)},q^{(2)}}(\omega)^* M_{\widetilde{p},\widetilde{q}^{(1)},\widetilde{q}^{(2)}}(\omega) = I_2, \ \omega \in \mathbb{R}.$$

Throughout this paper, M^* denotes the complex conjugate and transpose of a matrix M.

For a pair of FIR frame filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$, let ϕ and $\widetilde{\phi}$ denote the associated refinable (or scaling) functions satisfying the refinement equations

$$\phi(x) = \sum_k p_k \phi(2x-k), \ \widetilde{\phi}(x) = \sum_k \widetilde{p}_k \widetilde{\phi}(2x-k),$$

and let $\psi^{(\ell)}, \widetilde{\psi}^{(\ell)}, \ell = 1, 2$ be the functions defined by

$$\psi^{(\ell)}(x) = \sum_k q^{(\ell)}_k \phi(2x-k), \ \widetilde{\psi}^{(\ell)}(x) = \sum_k \widetilde{q}^{(\ell)}_k \widetilde{\phi}(2x-k).$$

We say that $\psi^{(\ell)}, \widetilde{\psi}^{(\ell)}, \ell = 1, 2$ generate **biorthogonal wavelet frames** (**bi-frames** for short) of $L^2(\mathbb{R})$ or **dual wavelet frames** of $L^2(\mathbb{R})$ if $\{\psi^{(1)}_{j,k}(x), \psi^{(2)}_{j,k}(x)\}_{j,k\in\mathbf{Z}}$ and $\{\widetilde{\psi}^{(1)}_{j,k}(x), \widetilde{\psi}^{(2)}_{j,k}(x)\}_{j,k\in\mathbf{Z}}$ are frames of $L^2(\mathbb{R})$ and that for any $f \in L^2(\mathbb{R})$, f can be written as (in L^2-norm)

$$f = \sum_{\ell=1,2} \sum_{j,k \in \mathbf{Z}} \langle f, \widetilde{\psi}^{(\ell)}_{j,k} \rangle \psi^{(\ell)}_{j,k}.$$

The Mixed Unitary Extension Principle (**MUEP**) of [48] (also see [21]) states that if $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ are biorthogonal, $\phi, \widetilde{\phi} \in L^2(\mathbb{R})$ with $\widehat{\phi}(0)\widehat{\widetilde{\phi}}(0) \neq 0$, and that $p(0) = \widetilde{p}(0) = 1$, $p(\pi) = \widetilde{p}(\pi) = q^{(\ell)}(0) = \widetilde{q}^{(\ell)}(0) = 0$, then $\psi^{(\ell)}, \widetilde{\psi}^{(\ell)}, \ell = 1, 2$ generate bi-frames of $L^2(\mathbb{R})$.

For a frame filter bank $\{p, q^{(1)}, q^{(2)}\}$, when it is used as the analysis filter bank, the frame multiresolution decomposition algorithm for input data $\{c_k\}$ is

$$\widetilde{c}_n = \frac{1}{2} \sum_{k \in \mathbf{Z}} p_{k-2n} c_k, \ d_n^{(1)} = \frac{1}{2} \sum_{k \in \mathbf{Z}} q_{k-2n}^{(1)} c_k, \ d_n^{(2)} = \frac{1}{2} \sum_{k \in \mathbf{Z}} q_{k-2n}^{(2)} c_k. \tag{2}$$

If an FIR frame filter bank $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ is biorthogonal to $\{p, q^{(1)}, q^{(2)}\}$, then $\{c_k\}$ can be recovered from $\widetilde{c}_n$ and $d_n^{(1)}, d_n^{(2)}$:

$$c_k = \sum_{n \in \mathbf{Z}} \widetilde{p}_{k-2n} \widetilde{c}_n + \sum_{n \in \mathbf{Z}} \widetilde{q}_{k-2n}^{(1)} d_n^{(1)} + \sum_{n \in \mathbf{Z}} \widetilde{q}_{k-2n}^{(2)} d_n^{(2)}, \quad k \in \mathbf{Z}. \tag{3}$$

(3) is called the frame multiresolution reconstruction algorithm, and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ is called the (frame) synthesis filter bank. $\{\widetilde{c}_k\}_k$ is called the "approximation" of $\{c_k\}_k$, $\{d_k^{(1)}\}_k$ and $\{d_k^{(2)}\}_k$ the "detail" of $\{c_k\}_k$. $\{\widetilde{c}_k\}_k$ and $\{d_k^{(1)}\}_k$, $\{d_k^{(2)}\}_k$ are also called the lowpass output and highpass outputs of $\{c_k\}_k$ respectively.

When filter banks are used for surface multiresolution processing, two issues need to be addressed. The first one is that the algorithms should be given by templates so that the algorithms can be easily implemented. The second issue is the symmetry of the filters. Unlike an image, a set of 2-D data, a surface (mesh) is an object in 3-D space that consists of not only regular vertices but also extraordinary vertices in general, while the algorithms for surface processing are derived from 2-D filter banks and the algorithm templates are given in the 2-D parametric plane. Thus it is required that these algorithms and templates have high symmetry so that they can be easily implemented for surface processing and that one can design the corresponding algorithm templates for extraordinary vertices. The reader is referred to [2, 44, 45, 54, 56, 57] for surface multiresolution processing.

For open surfaces, besides multiresolution algorithms for interior vertices on these surfaces, special algorithms are designed to process boundary vertices, see e.g. [57]. These special algorithms for boundary vertices can be derived from 1-D wavelets or frames. When 1-D bi-frames are used as boundary algorithms, we also need to consider the two issues mentioned above: template representation and symmetry. For the first issue, using the idea in our recent work [42, 41], where templates of multiresolution algorithms derived from 2-D wavelets are obtained, we will have the corresponding algorithm templates when we appropriately associate $\widetilde{c}_k, d_k^{(1)}, d_k^{(2)}$ with the nodes of $\mathbf{Z}$. For the symmetry issue, it is required that all the 1-D algorithm templates of the lowpass and highpass analysis algorithms and the synthesis algorithm be symmetric, or equivalently, not only $\phi, \widetilde{\phi}$, but also all $\psi^{(\ell)}, \widetilde{\psi}^{(\ell)}, \ell = 1, 2$ are symmetric. We say a frame has **uniform symmetry** if its associated refinable function and each of its framelets are symmetric.

The construction of 1-D wavelet tight frames and bi-framelets has been studied in many papers, see e.g., [10, 11, 12, 21, 22, 31, 33, 39, 50, 51, 52]. However, not all framelets are symmetric. Except a few framelets in [12, 21], to the author's best knowledge, at least one of the constructed 1-D framelets in the literature is antisymmetric. While the uniformly symmetric framelets in [12, 21] are constructed by the Mixed Oblique Extension Principle (**MOEP**) that is based a vanishing moment recovery function (or a fundamental function of the parent vectors). The MOEP-based framelets result in multiresolution algorithms not as simple as the MUEP-based algorithms in (2) and (3). On the other hand, a small size of algorithm templates is critical for curve/surface multiresolution processing. Thus, we choose to use MUEP for the construction, and we will start with symmetric templates of small size (as small as possible) with the templates given by some parameters. Then we select the parameters such that the resulting framelets have optimal smoothness and vanishing moments. If the templates with a particular size cannot yield desired framelets, then we consider templates with a bigger size. Since the templates are symmetric, the resulting frames have uniform symmetry. The constructed symmetric bi-frames are optimal in the sense that with templates of particular (small) sizes, they achieve the highest smoothness and/or vanishing moment orders.

The lifting scheme is a powerful method to construct biorthogonal filter banks, see [55, 18]. Recently, based on the lifting scheme method, biorthogonal wavelets with high symmetry for surface multiresolution processing have been constructed in [2, 56, 57, 42, 41]. In this paper use the lifting scheme to construct bi-frames. More precisely, the procedure of our construction is that first we start with symmetric templates of the decomposition and reconstruction algorithms. These algorithm templates are given by several iterative steps with each step given by a template (the idea of the lifting scheme is used in this stage of our procedure). Then we obtain the corresponding bi-frame filter banks that are given by some parameters. Finally, we select the parameters based on the smoothness and vanishing moments of framelets.

Different ways to associate $\widetilde{c}_k$ and $d_k^{(1)}, d_k^{(2)}$ with the nodes of $\mathbf{Z}$ will result in different templates for the decomposition algorithm (2) and the reconstruction algorithm (3). In this paper we give two ways of the association that result in two types of frames, called type I and type II frames respectively. To provide a clearer picture on our procedure for bi-frame construction, we first consider a similar procedure for the template-based construction of biorthogonal wavelets. The rest of this paper is organized as follows. In §2, we show how the association of biorthogonal wavelet lowpass and highpass outputs to the nodes of $\mathbf{Z}$ results in multiresolution algorithm templates, and discuss how to get the biorthogonal filter banks corresponding to given multiresolution algorithm templates. The construction of bi-frames of type I and type II are investigated in §3 and §4 respectively. In §4, we also construct 4-point interpolatory subdivision scheme-based bi-frames.

2 Biorthogonal wavelets and associated multiresolution algorithm templates

FIR filter banks $\{p, q\}$ and $\{\widetilde{p}, \widetilde{q}\}$ are said to be biorthogonal or they are perfect reconstruction (PR) filter banks if they satisfy the biorthogonal conditions:

$$\begin{cases} \overline{p(\omega)}\widetilde{p}(\omega) + \overline{p(\omega+\pi)}\widetilde{p}(\omega+\pi) = 1, \\ \overline{p(\omega)}\widetilde{q}(\omega) + \overline{p(\omega+\pi)}\widetilde{q}(\omega+\pi) = 0, \ \omega \in \mathbb{R} \\ \overline{q(\omega)}\widetilde{q}(\omega) + \overline{q(\omega+\pi)}\widetilde{q}(\omega+\pi) = 1. \end{cases} \tag{4}$$

Suppose lowpass filters p and $\widetilde{p}$ satisfy the first equation in (4). Let q and $\widetilde{q}$ be the highpass filters given by $q_n = (-1)^{n-1}\widetilde{p}_{1-n}$ and $\widetilde{q}_n = (-1)^{n-1}p_{1-n}$. Then $\{p, q\}$ and $\{\widetilde{p}, \widetilde{q}\}$ are biorthogonal, see [17].

The multiresolution decomposition algorithm with an analysis filter bank $\{p, q\}$ for input data

$\{c_k\}$ is

$$\widetilde{c}_n = \frac{1}{2}\sum_{k\in\mathbf{Z}} p_{k-2n}c_k, \quad d_n = \frac{1}{2}\sum_{k\in\mathbf{Z}} q_{k-2n}c_k. \tag{5}$$

When the synthesis filter bank $\{\widetilde{p}, \widetilde{q}\}$ is biorthogonal to $\{p, q\}$, then $\{c_k\}$ can be recovered from $\widetilde{c}_n$ and d_n by the multiresolution reconstruction algorithm:

$$c_k = \sum_{n\in\mathbf{Z}} \widetilde{p}_{k-2n}\widetilde{c}_n + \sum_{n\in\mathbf{Z}} \widetilde{q}_{k-2n}d_n, \quad k \in \mathbf{Z}. \tag{6}$$

$\{\widetilde{c}_k\}_k$, $\{d_k\}_k$ are called the "approximation" (or "lowpass output") and the "detail" (or "highpass output") of $\{c_k\}_k$ respectively. The decomposition algorithm can be applied to the "approximation" $\{\widetilde{c}_n\}_n$ to get the "approximation" and "detail" of $\{\widetilde{c}_n\}_n$. The reconstruction algorithm then recovers $\{\widetilde{c}_n\}_n$ from its (coarsest) "approximation" and "details".

When $d_n = 0$, then (6) is reduced to $\hat{c}_k = \sum_{n\in\mathbf{Z}} \widetilde{p}_{k-2n}\widetilde{c}_n$. This is the subdivision algorithm with subdivision mask $\{\widetilde{p}_k\}$ to produce a finer polygon with vertices $\hat{c}_k$ from a coarse polygon with vertices $\widetilde{c}_k$.

Let $p(\omega) = \frac{1}{2}\sum_k p_k e^{-ik\omega}$ be an FIR lowpass filter. We say $p(\omega)$ has **sum rule order** M if

$$p(0) = 1, \ \left.\frac{d^j}{d\omega^j}p(\omega)\right|_{\omega=\pi} = 0, \tag{7}$$

for $j = 0, 1, \cdots, M-1$. Assume that $p(\omega)$ is supported on $[-K, K]$, namely, $p_k = 0$ for $|k| > K$, where K is a positive integer. Let T_p be the transition operator matrix defined by

$$T_p = \left[A_{2k-j}\right]_{k,j\in[-K,K]}, \tag{8}$$

where $A_j = \frac{1}{2}\sum_{n\in\mathbf{Z}} p_{n-j}p_n$. We say T_p to satisfy *Condition E* if 1 is its simple eigenvalue and all other eigenvalues λ of T_p satisfy $|\lambda| < 1$.

Suppose $\{p, q\}$ and $\{\widetilde{p}, \widetilde{q}\}$ are a pair of biorthogonal FIR filter banks. Then from the integer-shift invariant multiresolution analysis theory (see e.g. [36]), if $p, \widetilde{p}$ have sum rule of order at least 1, and that the transition operator matrices T_p and $T_{\widetilde{p}}$ associated with p and $\widetilde{p}$ satisfy *Condition E*, then ϕ and $\widetilde{\phi}$ are biorthogonal duals: $\int_{\mathbb{R}} \phi(x)\overline{\widetilde{\phi}(x-k)}\,dx = \delta_k, k \in \mathbf{Z}$. Furthermore, $\psi, \widetilde{\psi}$, defined by $\hat{\psi}(\omega) = q(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2}), \hat{\widetilde{\psi}}(\omega) = \widetilde{q}(\frac{\omega}{2})\hat{\widetilde{\phi}}(\frac{\omega}{2})$ are biorthogonal wavelets, namely, $\{\psi_{j,k}\}_{j,k\in\mathbf{Z}}$ and $\{\widetilde{\psi}_{j,k}\}_{j,k\in\mathbf{Z}}$ are biorthogonal bases of $L^2(\mathbb{R})$. Throughout this paper, $\hat{f}$ denotes the Fourier transform of a function f on $\mathbb{R}$.

A subdivision algorithm can be given by templates (or stencils) so that the algorithm can be easily implemented. It is desirable that the multiresolution algorithms, which involve not only lowpass filters but also highpass filters, should be represented by some templates. The key for this is to associate appropriately $\widetilde{c}_k, d_k$, with the nodes of $\mathbf{Z}$. Next, we describe the association.

Figure 1: *Left: Original data $\{v_k, e_k\}$; Right: Decomposed data $\{\widetilde{v}_k\}$ and $\{\widetilde{e}_k\}$*

For initial data $\{c_k\}$, denote

$$v_k = c_{2k}, \ e_k = c_{2k+1}, \ k \in \mathbf{Z}. \tag{9}$$

v_k and e_k are shown on the left of Fig. 1. Let $\{\widetilde{c}_k\}$ and $\{d_k\}$ be the lowpass and highpass outputs with a filter bank $\{p, q\}$. Denote

$$\widetilde{v}_k = \widetilde{c}_k,\ \widetilde{e}_k = d_k,\ k \in \mathbf{Z}. \tag{10}$$

Thus the decomposition algorithm is to obtain $\widetilde{v}$ and $\widetilde{e}$ from $\{v, e\}$, while the reconstruction algorithm is to obtain $\{v, e\}$ from $\widetilde{v}$ and $\widetilde{e}$. If we associate $\widetilde{v}_k$ and $\widetilde{e}_k$ with nodes $2k$ and $2k+1$ respectively (see the right of Fig. 1), then the decomposition algorithm and the reconstruction algorithm can be given by templates. In the following, as an example, let us give the templates for the multiresolution algorithms with a pair of biorthogonal filter banks from [17].

Let $\{p, q\}$ and $\{\widetilde{p}, \widetilde{q}\}$ be the pair of biorthogonal filter banks in [17] with nonzero coefficients $p_k, \widetilde{p}_k, q_k, \widetilde{q}_k$ given by

$$\begin{aligned}
&[p_{-1}, p_0, p_1] = [\frac{1}{2}, 1, \frac{1}{2}], \quad [\widetilde{p}_{-2}, \cdots, \widetilde{p}_2] = [-\frac{1}{4}, \frac{1}{2}, \frac{3}{2}, \frac{1}{2}, -\frac{1}{4}],\\
&[q_{-1}, q_0, \cdots, q_3] = [-\frac{1}{4}, -\frac{1}{2}, \frac{3}{2}, -\frac{1}{2}, -\frac{1}{4}], \quad [\widetilde{q}_0, \widetilde{q}_1, \widetilde{q}_2] = [-\frac{1}{2}, 1, -\frac{1}{2}].
\end{aligned}$$

Then the decomposition algorithm with $\{p, q\}$ is

$$\begin{aligned}
\widetilde{c}_n &= \tfrac{1}{4}c_{2n-1} + \tfrac{1}{2}c_{2n} + \tfrac{1}{4}c_{2n+1},\\
d_n &= -\tfrac{1}{8}c_{2n-1} - \tfrac{1}{4}c_{2n} + \tfrac{3}{4}c_{2n+1} - \tfrac{1}{4}c_{2n+2} - \tfrac{1}{8}c_{2n+3},
\end{aligned} \tag{11}$$

and the reconstruction algorithm with $\{\widetilde{p}, \widetilde{q}\}$ can be written as

$$\begin{aligned}
c_{2k} &= -\tfrac{1}{4}\widetilde{c}_{k-1} + \tfrac{3}{2}\widetilde{c}_k - \tfrac{1}{4}\widetilde{c}_{k+1} - \tfrac{1}{2}d_{k-1} - \tfrac{1}{2}d_k,\\
c_{2k+1} &= \tfrac{1}{2}\widetilde{c}_k + \tfrac{1}{2}\widetilde{c}_{k+1} + d_k.
\end{aligned} \tag{12}$$

With the notations in (9) and (10), the decomposition algorithm (11) can be written as

$$\begin{aligned}
\widetilde{v}_n &= \tfrac{1}{4}e_{n-1} + \tfrac{1}{2}v_n + \tfrac{1}{4}e_n,\\
\widetilde{e}_n &= -\tfrac{1}{8}e_{n-1} - \tfrac{1}{4}v_n + \tfrac{3}{4}e_n - \tfrac{1}{4}v_{n+1} - \tfrac{1}{8}e_{n+1},
\end{aligned} \tag{13}$$

and the reconstruction algorithm (12) can be written as

$$\begin{aligned}
v_k &= -\tfrac{1}{4}\widetilde{v}_{k-1} + \tfrac{3}{2}\widetilde{v}_k - \tfrac{1}{4}\widetilde{v}_{k+1} - \tfrac{1}{2}\widetilde{e}_{k-1} - \tfrac{1}{2}\widetilde{e}_k,\\
e_k &= \tfrac{1}{2}\widetilde{v}_k + \tfrac{1}{2}\widetilde{v}_{k+1} + \widetilde{e}_k.
\end{aligned} \tag{14}$$

Thus, the decomposition algorithm (13) to obtain $\widetilde{v}_k$ $(= \widetilde{c}_k)$ can be represented as the template on the left of Fig. 2, and that to obtain $\widetilde{e}_k$ $(= d_k)$ can be represented as the template on right of Fig. 2. The reconstruction algorithm (14) can be represented as templates in Fig. 3 with the left template to recover v_k $(= c_{2k})$ and the right template to recover e_k $(= c_{2k+1})$. In Figs. 2 and 3, we have given the templates for $k = 0$. For the other values of k, the templates are the same except for the changes of indices.

As mentioned in the introduction, symmetry of the algorithms (templates) is important for curve/surface multiresolution processing. The above biorthogonal filter banks do result in symmetric templates as shown in Figs. 2 and 3. Actually, the biorthogonal filter banks with ϕ and $\widetilde{\phi}$ symmetric around the origin 0 in [17] also result in symmetric templates. Thus the templates derived from these biorthogonal filters can be used to process the boundary vertices in surface multiresolution processing.

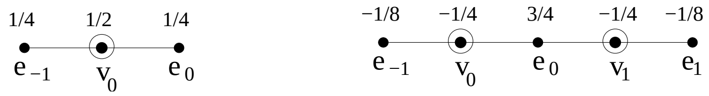

Figure 2: *Left: Template for decomposition algorithm to get $\widetilde{v}_k$ (for $k=0$); Right: Template for decomposition algorithm to get $\widetilde{e}_k$ (for $k=0$)*

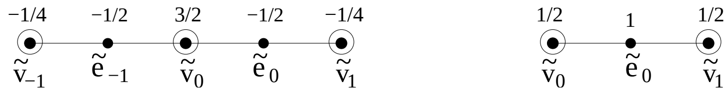

Figure 3: *Templates for reconstruction algorithm to recover v_k (left) and to recover e_k (right) for $k=0$*

As mentioned above, our frame algorithms are given by iterative steps with each step of algorithm being represented by a template. Since a frame has in general two or more generators (it has two generators in this paper), more highpass filters are involved than biorthogonal wavelets. Thus, the template-based construction of biorthogonal symmetric wavelets will give us a clearer idea about our method for the construction of bi-frames. Next, let us consider a 2-step (biorthogonal wavelet) multiresolution algorithm.

2-step Decomposition Algorithm:

Step 1. $\widetilde{v} = \frac{1}{b}\{v - d(e_{-1} + e_0)\};$ (15)

Step 2. $\widetilde{e} = e - u(\widetilde{v}_0 + \widetilde{v}_1).$ (16)

2-step Reconstruction Algorithm:

Step 1. $e = \widetilde{e} + u(\widetilde{v}_0 + \widetilde{v}_1);$ (17)

Step 2. $v = b\widetilde{v} + d(e_{-1} + e_0).$ (18)

Figure 4: *Left: Decomposition Step 1; Right: Decomposition Step 2*

The decomposition algorithm is given in (15) and (16) and shown in Fig. 4, where b, d, u are some constants. Namely, first we replace each v associated with an even node $2k$ by $\widetilde{v}$ given in (15). After that with the obtained $\widetilde{v}$, we update e associated with an odd node $2k+1$ by $\widetilde{e}$ given in (16). The algorithm to obtain the lowpass output $\widetilde{v}$ and highpass output $\widetilde{e}$ is very simple.

The reconstruction algorithm is given in (17) and (18) and shown in Fig. 5, where b, d, u are the same constants in the decomposition algorithm. More precisely, first we replace each $\widetilde{e}$ of the highpass output by e given in (17). This step recovers original data c_{2k+1} associated with odd nodes. After that, with the obtained e, we replace $\widetilde{v}$ of the lowpass output by v with the formula given in (18). This step recovers original data c_{2k} associated with even nodes. The reconstruction algorithm is also very simple.

To choose the constants b, d, u, we need to study the properties of the corresponding wavelets. To this regard, we first need to obtain the corresponding biorthogonal filter banks, denoted as

Figure 5: *Left: Reconstruction Step 1; Right: Reconstruction Step 2*

$\{_2p, {_2q}\}$, and $\{_2\widetilde{p}, {_2\widetilde{q}}\}$. In the following, let us give the details about how to get $\{_2p, {_2q}\}$, and $\{_2\widetilde{p}, {_2\widetilde{q}}\}$.

From (15), we have

$$\widetilde{c}_n = \widetilde{v}_n = \frac{1}{b}\{v_n - d(e_{n-1} + e_n)\} = \frac{1}{b}\{c_{2n} - d(c_{2n-1} + c_{2n+1})\}. \tag{19}$$

This and (16) imply

$$\begin{aligned} d_n &= \widetilde{e}_n = e_n - u(\widetilde{v}_n + \widetilde{v}_{n+1}) \\ &= c_{2n+1} - u(\frac{1}{b}\{c_{2n} - d(c_{2n-1} + c_{2n+1})\} + \frac{1}{b}\{c_{2n+2} - d(c_{2n+1} + c_{2n+3})\}) \\ &= (1 + \frac{2ud}{b})c_{2n+1} - \frac{u}{b}(c_{2n} + c_{2n+2}) - \frac{ud}{b}(c_{2n+1} + c_{2n+3}). \end{aligned} \tag{20}$$

Thus, comparing (19) and (20) with (5), we get the nonzero coefficients $_2p_k, {_2q_k}$ of $_2p(\omega), {_2q(\omega)}$:

$$\begin{aligned} &_2p_0 = \tfrac{2}{b},\ {_2p_{-1}} = {_2p_1} = -\tfrac{2d}{b}; \\ &_2q_0 = 2(1 + \tfrac{2ud}{b}),\ {_2q_0} = {_2q_1} = -\tfrac{2u}{b},\ {_2q_{-1}} = {_2q_3} = -\tfrac{2ud}{b}. \end{aligned}$$

Therefore, the analysis filter bank is

$$_2p(\omega) = \frac{1}{b} - \frac{d}{b}(e^{i\omega} + e^{-i\omega}),\ {_2q(\omega)} = (1 + \frac{2ud}{b})e^{-i\omega} - \frac{u}{b}(1 + e^{-i2\omega}) - \frac{ud}{b}(e^{i\omega} + e^{-i3\omega}).$$

Next, let us obtain the synthesis filter bank $\{_2\widetilde{p}, {_2\widetilde{q}}\}$. From (17), we have

$$c_{2k+1} = e_k = \widetilde{e}_k + u(\widetilde{v}_k + \widetilde{v}_{k+1}) = d_k + u(\widetilde{c}_k + \widetilde{c}_{k+1}). \tag{21}$$

This and (18) lead to

$$\begin{aligned} c_{2k} &= v_k = b\widetilde{v}_k + d(e_{k-1} + e_k) \\ &= b\widetilde{c}_k + d\{d_{k-1} + u(\widetilde{c}_{k-1} + \widetilde{c}_k) + d_k + u(\widetilde{c}_k + \widetilde{c}_{k+1})\} \\ &= (b + 2du)\widetilde{c}_k + du(\widetilde{c}_{k-1} + \widetilde{c}_{k+1}) + d(d_{k-1} + d_k). \end{aligned} \tag{22}$$

Thus, comparing (21) with (6) for odd k, we get that

$$_2\widetilde{p}_1 = {_2\widetilde{p}_{-1}} = u,\ {_2\widetilde{q}_1} = 1,$$

and the other coefficients $_2\widetilde{p}_{2k+1}, {_2\widetilde{q}_{2k+1}}$ with odd indices $2k+1$ are zero, while comparing (22) with (6) for even k, we have that the nonzero $_2\widetilde{p}_{2k}, {_2\widetilde{q}_{2k}}$ with even indices $2k$ are

$$_2\widetilde{p}_0 = b + 2du,\ {_2\widetilde{p}_2} = {_2\widetilde{p}_{-2}} = du,\ {_2\widetilde{q}_2} = {_2\widetilde{q}_0} = d.$$

Hence, the synthesis filter bank is

$$_2\widetilde{p}(\omega) = \frac{1}{2}(b + 2du) + \frac{u}{2}(e^{-i\omega} + e^{i\omega}) + \frac{du}{2}(e^{-i2\omega} + e^{i2\omega}),\ {_2\widetilde{q}(\omega)} = \frac{1}{2}e^{-i\omega} + \frac{d}{2}(1 + e^{-i2\omega}).$$

Denote

$$A_1(\omega) = \begin{bmatrix} 1 & 0 \\ -u(1+e^{-i\omega}) & 1 \end{bmatrix}, \; A_0(\omega) = \begin{bmatrix} \frac{1}{b} & -\frac{d}{b}(1+e^{i\omega}) \\ 0 & 1 \end{bmatrix}, \tag{23}$$

$$\widetilde{A}_1(\omega) = \begin{bmatrix} 1 & u(1+e^{i\omega}) \\ 0 & 1 \end{bmatrix}, \; \widetilde{A}_0(\omega) = \begin{bmatrix} b & 0 \\ d(1+e^{-i\omega}) & 1 \end{bmatrix}. \tag{24}$$

Then $\{_2p, {_2q}\}$ and $\{_2\widetilde{p}, {_2\widetilde{q}}\}$ can be written as

$$\begin{bmatrix} _2p(\omega) \\ _2q(\omega) \end{bmatrix} = A_1(2\omega)A_0(2\omega)\begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix}, \quad \begin{bmatrix} _2\widetilde{p}(\omega) \\ _2\widetilde{q}(\omega) \end{bmatrix} = \frac{1}{2}\widetilde{A}_1(2\omega)\widetilde{A}_0(2\omega)\begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix}.$$

The example above shows how to find biorthogonal filter banks corresponding to templates of a multiresolution algorithm. In the following, we construct biorthogonal filters and bi-frame filters with algorithms given by templates similar to that in Figs. 4 and 5. The corresponding filter banks can be obtained similarly as we do above with $\{_2p, {_2q}\}$ and $\{_2\widetilde{p}, {_2\widetilde{q}}\}$. The filters are given by some parameters. We then choose the parameters based on the smoothness and vanishing moments of framelets.

For an FIR (highpass) filter $q(\omega)$, we say it has **vanishing moments of order** J if

$$\frac{d^j}{d\omega^j}q(\omega)\Big|_{\omega=0} = 0, \quad 0 \le j < J.$$

Clearly, if $q(\omega)$ has vanishing moment order J and ψ is the compactly supported function defined by $\hat{\psi}(\omega) = q(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2})$, where ϕ is a compactly supported function in $L^2(\mathbb{R})$, then ψ has vanishing moments of order J:

$$\int_{-\infty}^{\infty} \psi(x)x^j dx = 0, \quad 0 \le j < J.$$

Most importantly, one can show that if $q(\omega)$ has vanishing moment order J, then when it is used as the analysis highpass filter, it annihilates discrete polynomials of degree less than J, namely, when $c_k = P(k)$, where P is a polynomial with degree $< J$, then

$$d_n = \frac{1}{2}\sum_{k\in\mathbf{Z}} q_{k-2n}P(k) = 0, \quad n \in \mathbf{Z}.$$

It is important in signal/image processing and other applications that highpass filters annihilate discrete polynomials.

When we consider the smoothness of wavelets/framelets, we consider the Sobolev smoothness in this paper. For $s \ge 0$, let W^s denote the Sobolev space consisting of functions $f(x)$ on $\mathbb{R}$ with $\int_{\mathbb{R}}(1+|\omega|^2)^s|\hat{f}(\omega)|^2 d\omega < \infty$. Clearly, if $f \in W^s$ with $s > \frac{1}{2}$, then f is in the Hölder space $C^{s-\frac{1}{2}-\epsilon}$ for any $\epsilon > 0$. The Sobolev smoothness of a scaling function ϕ can be given by the eigenvalues of T_p, where p is the associated lowpass filter. More precisely, assume that $p(\omega)$ has sum rule order m. Denote $S_m = \mathrm{spec}(T_p)\backslash\{1, \frac{1}{2}, \frac{1}{4}, \cdots, \frac{1}{2^{2m-1}}\}$, and $\rho_0 = \max\{|\lambda| : \lambda \in S_m\}$. Then ϕ is in Sobolev space $W^{-\log_2 \rho_0 - \epsilon}$ for any $\epsilon > 0$, see [25, 58]. See also [38, 37] for similar formulas for the Sobolev smoothness of high-dimensional and multiple scaling functions.

When we construct biorthogonal wavelets and bi-frames, we choose the parameters such that the synthesis scaling function $\widetilde{\phi}$ is smoother than the analysis scaling function ϕ, and that the analysis highpass filters have higher vanishing moments. One can easily verify that for a pair biorthogonal of filter banks $\{p, q\}$ and $\{\widetilde{p}, \widetilde{q}\}$, q has vanishing moments order J if and only if $\widetilde{p}$ has sum rule order J. Thus when we construct biorthogonal wavelets, we choose the parameters such that $\widetilde{p}$ has a higher sum rule order than p (hence, q has a higher vanishing moment order than $\widetilde{q}$).

About the selection of the values for the parameters, we first solve the system of linear equations for sum rule orders of lowpass filters and for the vanishing moments of highpass filters. (The orders of sum rule and vanishing moments depend on the algorithms.) After that we select the remaining parameters such that $\widetilde{\phi}$ and/or ϕ have the optimal Sobolev smoothness by minimizing $\widetilde{\rho}_0$ for $\widetilde{\phi}$ (ρ_0 for ϕ). One could use the Matlab function `fmincon` for minimization.

Now let us return back to the above 2-step multiresolution algorithm. Solving the system of equations for sum rule order 1 of both ${}_2p$ and ${}_2\widetilde{p}$, we obtain

$$b = 2,\ d = -\frac{1}{2},\ u = \frac{1}{2}.$$

The resulting ${}_2p, {}_2\widetilde{p}$ actually have sum rule order 2. More precisely, they are

$${}_2p(\omega) = \frac{1}{4}e^{i\omega}(1 + e^{-i\omega})^2,\ {}_2\widetilde{p}(\omega) = \frac{1}{8}(-1 + 4e^{i\omega} - e^{i2\omega})(1 + e^{-i\omega})^2. \tag{25}$$

Thus the resulting ϕ is the linear B-spline supported on $[-1, 1]$. Using the smoothness formula provided above, one can obtain $\widetilde{\phi} \in W^{0.44076}$. To obtain a smoother $\widetilde{\phi}$, we need to consider algorithms with more steps. In the following two examples, we consider 3-step and 4-step algorithms. As the 2-step algorithm, the decomposition algorithm of each of these two algorithms is to obtain lowpass output $\widetilde{v}$ and highpass output $\widetilde{e}$ from input $\{v, e\}$, and the reconstruction algorithm is to recover $\{v, e\}$ from both $\widetilde{v}$ and $\widetilde{e}$.

Example 1. *In this example, we consider a 3-step multiresolution algorithm. The decomposition algorithm is given in (26)-(28) and shown in Fig. 6, where b, d, u, d_1, c_1 are some constants. More precisely, first we replace each v associated with an even node $2k$ by v'' given in (26). After that with the obtained v'', we update e associated with an odd node $2k+1$ by $\widetilde{e}$ with the formula given in (27). Finally, v'' obtained in Step 1 is replaced by $\widetilde{v}$ given in (28).*

3-step Decomposition Algorithm:

Step 1. $v'' = \frac{1}{b}\{v - d(e_{-1} + e_0)\};$ (26)

Step 2. $\widetilde{e} = e - u(v''_0 + v''_1);$ (27)

Step 3. $\widetilde{v} = v'' - d_1(\widetilde{e}_{-1} + \widetilde{e}_0) - c_1(\widetilde{e}_{-2} + \widetilde{e}_1).$ (28)

3-step Reconstruction Algorithm:

Step 1. $v'' = \widetilde{v} + d_1(\widetilde{e}_{-1} + \widetilde{e}_0) + c_1(\widetilde{e}_{-2} + \widetilde{e}_1);$ (29)

Step 2. $e = \widetilde{e} + u(v''_0 + v''_1);$ (30)

Step 3. $v = bv'' + d(e_{-1} + e_0).$ (31)

The reconstruction algorithm is given in (29)-(31) and shown in Fig. 7, where b, d, u, d_1, c_1 are the same constants as in the decomposition algorithm. That is, first we replace each $\widetilde{v}$ of the

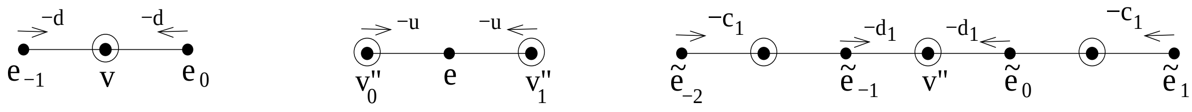

Figure 6: *Left: Decomposition Step 1; Middle: Decomposition Step 2; Left: Decomposition Step 3*

Figure 7: *Left: Reconstruction Step 1; Middle: Reconstruction Step 2; Right: Reconstruction Step 3*

lowpass output by v'' given in (29). Then, with the obtained v'', we update $\widetilde{e}$ of the highpass output by e given in (30). Finally, with the obtained e, we update v'' obtained in Step 1 by v given in (31).

Just as we obtained $\{{}_2p, {}_2q\}$ and $\{{}_2\widetilde{p}, {}_2\widetilde{q}\}$ above, we can similarly obtain the filter banks corresponding to algorithm (26)-(31). The filter banks, denoted as $\{{}_3p, {}_3q\}$ and $\{{}_3\widetilde{p}, {}_3\widetilde{q}\}$, are given by

$$\begin{bmatrix} {}_3p(\omega) \\ {}_3q(\omega) \end{bmatrix} = A_2(2\omega)A_1(2\omega)A_0(2\omega)\begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix}, \quad \begin{bmatrix} {}_3\widetilde{p}(\omega) \\ {}_3\widetilde{q}(\omega) \end{bmatrix} = \tfrac{1}{2}\widetilde{A}_2(2\omega)\widetilde{A}_1(2\omega)\widetilde{A}_0(2\omega)\begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$

where A_1, A_0 and $\widetilde{A}_1, \widetilde{A}_0$ are defined by (23) and (24) respectively, and

$$A_2(\omega) = \begin{bmatrix} 1 & -d_1(1+e^{i\omega}) - c_1(e^{-i\omega}+e^{i2\omega}) \\ 0 & 1 \end{bmatrix}, \; \widetilde{A}_2(\omega) = \begin{bmatrix} 1 & 0 \\ d_1(1+e^{-i\omega}) + c_1(e^{i\omega}+e^{-i2\omega}) & 1 \end{bmatrix}. \tag{32}$$

There are 5 parameters b, d, u, d_1, c_1 for $\{{}_3p, {}_3q\}$ and $\{{}_3\widetilde{p}, {}_3\widetilde{q}\}$. If we solve the system of equations for sum rule order 2 of ${}_3p$ and sum rule order 4 of ${}_3\widetilde{p}$, we have

$$b = \frac{1}{2},\ d = \frac{1}{4},\ u = \frac{1}{2},\ d_1 = -\frac{3}{8} - c_1.$$

In this case, ${}_3\widetilde{p}(\omega) = \frac{1}{16}e^{i2\omega}(1+e^{-i\omega})^4$. Thus $\widetilde{\phi}$ is the cubic C^2 B-spline supported on $[-2, 2]$. If we choose $c_1 = \frac{5}{64}$, then ${}_3p(\omega)$ has sum rule order 4, but the corresponding ϕ is not in $L^2(\mathbb{R})$. The best (numerically) smooth ϕ is that $\phi \in W^{0.13264}$. If we choose $c_1 = \frac{31}{256}$, then the resulting $\phi \in W^{0.13254}$; and $c_1 = \frac{1}{8}$, then $\phi \in W^{0.12976}$. In the following we provide the resulting ${}_3p, {}_3q, {}_3\widetilde{q}$ with $c_1 = \frac{1}{8}$ (hence, $d_1 = -\frac{1}{2}$):

$$\begin{aligned}
{}_3p(\omega) &= -\tfrac{e^{i\omega}}{32}\{e^{i4\omega}+e^{-i4\omega}-6(e^{i3\omega}+e^{-i3\omega})+13(e^{i2\omega}+e^{-i2\omega})-8(e^{i\omega}+e^{-i\omega})-8\}(1+e^{-i\omega})^2, \\
{}_3q(\omega) &= \tfrac{e^{i\omega}}{4}(1-e^{-i\omega})^4, \\
{}_3\widetilde{q}(\omega) &= \tfrac{1}{128}\{e^{i4\omega}+e^{-i4\omega}+6(e^{i3\omega}+e^{-i3\omega})+13(e^{i2\omega}+e^{-i2\omega})+8(e^{i\omega}+e^{-i\omega})-8\}(1-e^{-i\omega})^2.\diamondsuit
\end{aligned}$$

Example 2. *In this example, we consider a 4-step multiresolution algorithm. The decomposition and reconstruction algorithm is given by (33)-(36) and (37)-(40), and shown in Fig. 8 and Fig. 9 respectively, where b, d, u, d_1, c_1, u_1 are some constants.*

4-step Decomposition Algorithm:

Step 1. $v'' = \frac{1}{b}\{v - d(e_{-1} + e_0)\};$ (33)

$$\textit{Step 2. } e'' = e - u(v_0'' + v_1''); \tag{34}$$
$$\textit{Step 3. } \widetilde{v} = v'' - d_1(e_{-1}'' + e_0'') - c_1(e_{-2}'' + e_1''); \tag{35}$$
$$\textit{Step 4. } \widetilde{e} = e'' - u_1(\widetilde{v}_0 + \widetilde{v}_1). \tag{36}$$

4-step Reconstruction Algorithm:

$$\textit{Step 1. } e'' = \widetilde{e} + u_1(\widetilde{v}_0 + \widetilde{v}_1); \tag{37}$$
$$\textit{Step 2. } v'' = \widetilde{v} + d_1(e_{-1}'' + e_0'') + c_1(e_{-2}'' + e_1''); \tag{38}$$
$$\textit{Step 3. } e = e'' + u(v_0'' + v_1''); \tag{39}$$
$$\textit{Step 4. } v = bv'' + d(e_{-1} + e_0). \tag{40}$$

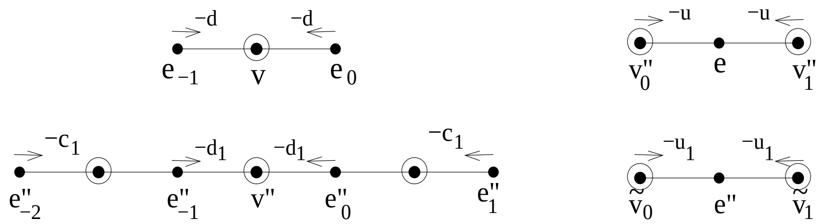

Figure 8: *Top-left: Decomposition Step 1; Top-right: Decomposition Step 2; Bottom-left: Decomposition Step 3; Bottom-right: Decomposition Step 4*

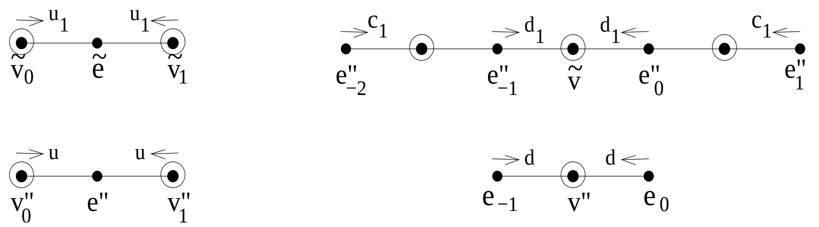

Figure 9: *Top-left: Reconstruction Step 1; Top-right: Reconstruction Step 2; Bottom-left: Reconstruction Step 3; Bottom-right: Reconstruction Step 4*

One can obtain the corresponding filter banks, denoted as $\{{}_4p, {}_4q\}$ *and* $\{{}_4\widetilde{p}, {}_4\widetilde{q}\}$*, to be*

$$\begin{bmatrix} {}_4p(\omega) \\ {}_4q(\omega) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -u_1(1+e^{-i2\omega}) & 1 \end{bmatrix} A_2(2\omega)A_1(2\omega)A_0(2\omega) \begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$
$$\begin{bmatrix} {}_4\widetilde{p}(\omega) \\ {}_4\widetilde{q}(\omega) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & u_1(1+e^{i2\omega}) \\ 0 & 1 \end{bmatrix} \widetilde{A}_2(2\omega)\widetilde{A}_1(2\omega)\widetilde{A}_0(2\omega) \begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$

where A_1 *and* A_0*,* $\widetilde{A}_1$ *and* $\widetilde{A}_0$*, and* A_2 *and* $\widetilde{A}_2$ *are defined by (23), (24) and (32) respectively.*

If we solve the system of equations for sum rule order 2 of ${}_4p$ *and sum rule order 4 of* ${}_4\widetilde{p}$*, we have (there are other solutions)*

$$b = \frac{1-2d}{1-2du_1-u_1},\ u = \frac{1-2u_1}{2(1-2du_1-u_1)},\ c_1 = \frac{1-4d-u_1+8d^2u_1+2du_1+4u_1^2-16d^2u_1^2}{64u_1^2(1-2d)},\ d_1 = -\frac{1}{4} - \frac{d}{2} - c_1,$$

Furthermore, if

$$d = \frac{1 - u_1 - \sqrt{1 - 6u_1 + 15u_1^2 - 20u_1^3 + 12u_1^4}}{4u_1(2 - 3u_1)}, \tag{41}$$

then ${}_4p(\omega)$ *has sum rule order 4. If we choose* $u_1 = \frac{5}{64}$*, then the resulting* $\widetilde{\phi} \in W^{3.40115}$*,* $\phi \in W^{0.00240}$*; while if we choose* $u_1 = \frac{3}{8}$*, then the resulting* $\widetilde{\phi}$ *and* ϕ *are in* $W^{2.51527}$ *and* $W^{0.89223}$ *respectively. When* $u_1 = \frac{3}{8}$*, the corresponding* b, d, u, d_1, c_1 *are*

$$b = -\frac{16}{39} + \frac{8\sqrt{43}}{39},\ d = \frac{10}{21} - \frac{\sqrt{43}}{42},\ u = \frac{15}{26} - \frac{\sqrt{43}}{26},\ d_1 = -\frac{647}{1008} + \frac{25\sqrt{43}}{1008},\ c_1 = \frac{155}{1008} - \frac{13\sqrt{43}}{1008}.$$

If we drop the condition (41) for sum rule order 4 of ${}_4p$*, we have two parameters* d, u_1*. It seems choosing different* d, u_1 *does not result in* $\widetilde{\phi}, \phi$ *with a significantly higher smoothness order. Here we provide two sets of* d, u_1*. With* $d = \frac{1}{4}, u_1 = \frac{1}{64}$*, the resulting* $\widetilde{\phi} \in W^{3.48584}$*,* $\phi \in W^{0.00771}$*, while* $d = u_1 = \frac{1}{4}$*, the resulting* $\widetilde{\phi} \in W^{2.82633}$*,* $\phi \in W^{0.86204}$*. The lowpass filters* $p(\omega)$ *and* $\widetilde{p}(\omega)$ *considered above are supported on* $[-5, 5]$ *and* $[-6, 6]$ *respectively.* $\diamondsuit$

One can obtain similarly the biorthogonal filter banks corresponding to the multiresolution algorithms with more iterative steps. Then, based on the filter banks, one can construct biorthogonal wavelets with higher smoothness orders and higher vanishing moment orders. Here we do not provide more examples.

3 Bi-frames with uniform symmetry: Type I

Suppose $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ are a pair of biorthogonal frame filter banks. Let $\widetilde{c}_k$ and $d_k^{(1)}, d_k^{(2)}$ be the lowpass output and highpass outputs of input c_k defined by (2) with the analysis frame filter bank $\{p, q^{(1)}, q^{(2)}\}$. For input $\{c_k\}$, as in §2, let $v_k = c_{2k}, e_k = c_{2k+1}$. Denote

$$\widetilde{v}_k = \widetilde{c}_k,\ \widetilde{f}_k = d_k^{(1)},\ \widetilde{e}_k = d_k^{(2)},\ k \in \mathbf{Z}. \tag{42}$$

Thus the frame decomposition algorithm is to obtain $\widetilde{v}$ and $\widetilde{f}, \widetilde{e}$ from $\{v, e\}$, while the frame reconstruction algorithm is to obtain $\{v, e\}$ from $\widetilde{v}$ and $\widetilde{f}, \widetilde{e}$. Associating $\widetilde{v}_k$ and $\widetilde{f}_k, \widetilde{e}_k$ with the nodes of $\mathbf{Z}$ appropriately, we can represent a frame multiresolution algorithm by templates.

Different ways to associate $\widetilde{v}_k$ and $\widetilde{f}_k, \widetilde{e}_k$ with the nodes of $\mathbf{Z}$ will result in different templates for the decomposition algorithm (2) and the reconstruction algorithm (3). Obviously, we should associate $\widetilde{c}_k$ with an even node. For $\widetilde{f}_k$ (one highpass output $d_k^{(1)}$) and $\widetilde{e}_k$ (the other highpass output $d_k^{(2)}$), we may associate both of them with an odd node, or one with an odd node but the other with an even node. These two ways of association result in two types of framelets, called type I and type II framelets in this paper.

The idea to construct uniformly symmetric framelets of either type I or type II is similar to that for biorthogonal wavelet construction discussed in §2. Namely, first we start with algorithms given by some templates of small sizes, then we find the the corresponding bi-frame filter banks, and finally, we choose the suitable parameters based on the smoothness and vanishing moments of the framelets. Bi-frames of type I and type II are investigated in this section and the next section respectively.

Before a specific frame algorithm is discussed, it shall be remarked that unlike the biorthogonal filters, $\widetilde{p}$ having a high sum rule order does not imply automatically $q^{(1)}$ and $q^{(2)}$ having high vanishing moment orders. Thus, when we design bi-frame filter banks, we need to solve not only the equations for the sum rule orders of $p, \widetilde{p}$, but also those for the vanishing moments of the highpass filters.

Next, let us consider a 2-step type I frame multiresolution algorithm. The decomposition algorithm is given in (43) and (44) and shown in Fig. 10, where b, d, u, w are some constants. Namely, first we replace each v associated with an even node $2k$ by $\widetilde{v}$ with the formula given in (43). After that with the obtained $\widetilde{v}$, we obtain the highpass outputs $\widetilde{f}, \widetilde{e}$ that are associated with odd nodes $2k+1$ by (44).

2-step Type I Frame Decomposition Algorithm:

Step 1. $\widetilde{v} = \frac{1}{b}\{v - d(e_{-1} + e_0)\};$ (43)

Step 2. $\widetilde{f} = e - u(\widetilde{v}_0 + \widetilde{v}_1), \ \widetilde{e} = e - w(\widetilde{v}_0 + \widetilde{v}_1).$ (44)

2-step Type I Frame Reconstruction Algorithm:

Step 1. $e = t\{\widetilde{f} + u(\widetilde{v}_0 + \widetilde{v}_1)\} + (1-t)\{\widetilde{e} + w(\widetilde{v}_0 + \widetilde{v}_1)\};$ (45)

Step 2. $v = b\widetilde{v} + d(e_{-1} + e_0).$ (46)

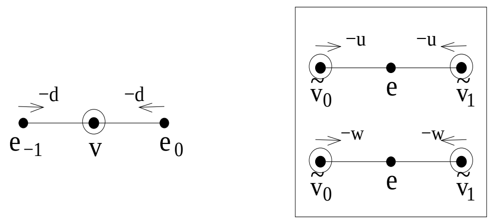

Figure 10: *Left: Decomposition Step 1; Right: Decomposition Step 2*

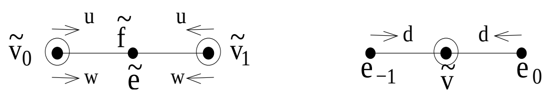

Figure 11: *Left: Reconstruction Step 1; Right: Reconstruction Step 2*

The reconstruction algorithm is given in (45) and (46) and shown in Fig. 11, where b, d, u, w are the same constants as in the decomposition algorithm and $t \in \mathbb{R}$. More precisely, first we obtain e, the original data c_{2k+1} associated with odd nodes, by a linear combination of $\widetilde{f}, \widetilde{e}, \widetilde{v}$ given by (45). After that, with the obtained e, we update $\widetilde{v}$ of the lowpass output by v with the formula given in (46). This step recovers original data c_{2k} associated with even nodes.

As in §2, one can obtain that the filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ corresponding to this 2-step frame algorithm are

$$\begin{aligned}
\left[p(\omega),\ q^{(1)}(\omega),\ q^{(2)}(\omega)\right]^T &= B_1(2\omega)B_0(2\omega)\left[1,\ e^{-i\omega}\right]^T, \\
\left[\widetilde{p}(\omega),\ \widetilde{q}^{(1)}(\omega),\ \widetilde{q}^{(2)}(\omega)\right]^T &= \tfrac{1}{2}\widetilde{B}_1(2\omega)\widetilde{B}_0(2\omega)\left[1,\ e^{-i\omega}\right]^T,
\end{aligned}$$

where

$$B_1(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ -u(1+e^{-i\omega}) & 1 & 0 \\ -w(1+e^{-i\omega}) & 0 & 1 \end{bmatrix}, \; B_0(\omega) = \begin{bmatrix} \frac{1}{b} & -\frac{d}{b}(1+e^{i\omega}) \\ 0 & 1 \\ 0 & 1 \end{bmatrix}, \tag{47}$$

$$\widetilde{B}_1(\omega) = \begin{bmatrix} 1 & u(1+e^{i\omega}) & w(1+e^{i\omega}) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; \widetilde{B}_0(\omega) = \begin{bmatrix} b & 0 \\ dt(1+e^{-i\omega}) & t \\ d(1-t)(1+e^{-i\omega}) & 1-t \end{bmatrix}. \tag{48}$$

There are 5 free parameters b, d, u, w, t. After solving the system of equations for sum rule order 1 of both $p, \widetilde{p}$ and for vanishing moment order 1 of $q^{(1)}, q^{(2)}$, this pair of filter banks is essentially reduced to $\{{}_2p, {}_2q\}$ and $\{{}_2\widetilde{p}, {}_2\widetilde{q}\}$ in §2 in the sense that the resulting $p, \widetilde{p}$ are ${}_2p, {}_2\widetilde{p}$ given in (25). Thus, this 2-step type I frame algorithm does not result in smoother $\phi, \widetilde{\phi}$ if $q^{(1)}, q^{(2)}$ have vanishing moment of order at least 1. To construct smoother $\phi, \widetilde{\phi}$, we need to consider algorithms with more iterative steps. Next we consider a 4-step bi-frame algorithm.

The decomposition algorithm of this 4-step algorithm is given in (49)-(52) and shown in Fig. 12, where $b, d, u, w, d_1, c_1, n_1, m_1, u_1, w_1$ are some constants. Namely, first we replace each v associated with an even node $2k$ by v'' given in (49). Then, with the obtained v'', we obtain f'', e'' associated with odd nodes $2k+1$ by (50). After that, v'' obtained in Step 1 is replaced by $\widetilde{v}$ given in (51). Finally, f'', e'' obtained in Step 2 are replaced by $\widetilde{f}, \widetilde{e}$ given in (52).

4-step Type I Frame Decomposition Algorithm:

Step 1. $v'' = \frac{1}{b}\{v - d(e_{-1} + e_0)\};$ (49)

Step 2. $f'' = e - u(v''_0 + v''_1), \; e'' = e - w(v''_0 + v''_1);$ (50)

Step 3. $\widetilde{v} = v'' - d_1(f''_{-1} + f''_0) - c_1(f''_{-2} + f''_1) - n_1(e''_{-1} + e''_0) - m_1(e''_{-2} + e''_1);$ (51)

Step 4. $\widetilde{f} = f'' - u_1(\widetilde{v}_0 + \widetilde{v}_1), \; \widetilde{e} = e'' - w_1(\widetilde{v}_0 + \widetilde{v}_1).$ (52)

4-step Type I Frame Reconstruction Algorithm:

Step 1. $f'' = \widetilde{f} + u_1(\widetilde{v}_0 + \widetilde{v}_1), \; e'' = \widetilde{e} + w_1(\widetilde{v}_0 + \widetilde{v}_1);$ (53)

Step 2. $v'' = \widetilde{v} + d_1(f''_{-1} + f''_0) + c_1(f''_{-2} + f''_1) + n_1(e''_{-1} + e''_0) + m_1(e''_{-2} + e''_1);$ (54)

Step 3. $e = t\{f'' + u(v''_0 + v''_1)\} + (1-t)\{e'' + w(v''_0 + v''_1)\};$ (55)

Step 4. $v = bv'' + d(e_{-1} + e_0).$ (56)

The reconstruction algorithm is given in (53)-(56) and shown in Fig. 13, where b, d, u, w, d_1, c_1, n_1, m_1, u_1, w_1 are the same constants as in the decomposition algorithm and $t \in \mathbb{R}$. First we replace each $\widetilde{f}, \widetilde{e}$ of the highpass outputs by f'', e'' respectively given in (53). Then, with the obtained f'', e'', we update $\widetilde{v}$ of the lowpass output by v'' in (54). After that, f'', e'' obtained in Step 1 are replaced by e with the formula given in (55). Finally, v'' obtained in Step 2 is replaced by v given in (56).

In the following, denote

$$z = e^{-i\omega}.$$

One can obtain the filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ corresponding to the above 4-step

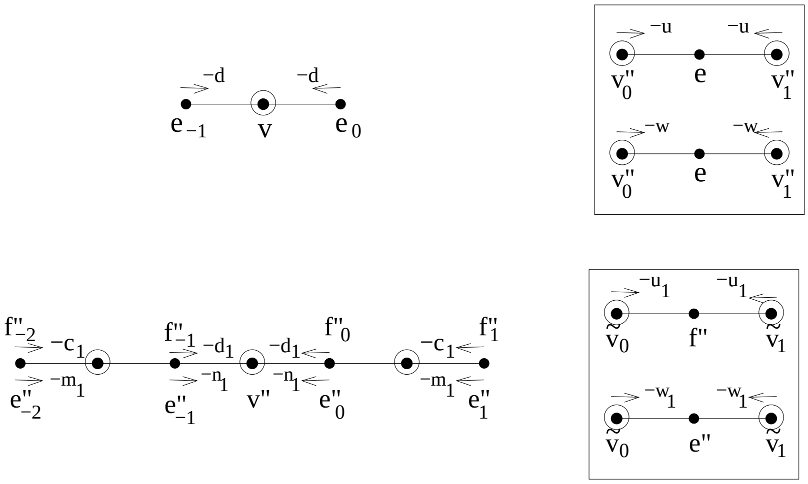

Figure 12: *Top-left: Decomposition Step 1; Top-right: Decomposition Step 2; Bottom-left: Decomposition Step 3; Bottom-right: Decomposition Step 4*

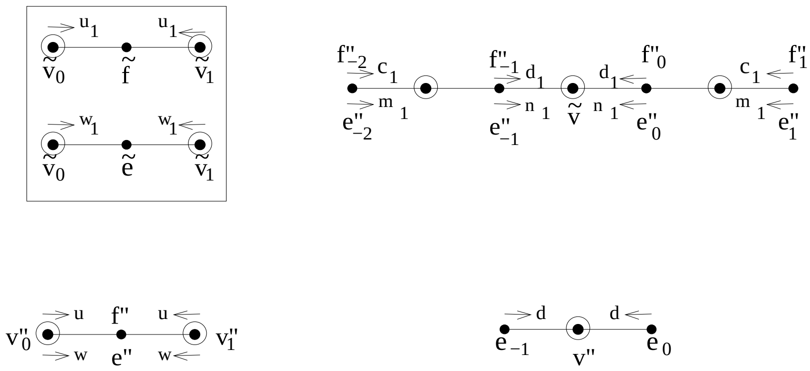

Figure 13: *Top-left: Reconstruction Step 1; Top-right: Reconstruction Step 2; Bottom-left: Reconstruction Step 3; Bottom-right: Reconstruction Step 4*

frame algorithm:

$$[p(\omega),\ q^{(1)}(\omega),\ q^{(2)}(\omega)]^T = \begin{bmatrix} 1 & 0 & 0 \\ -u_1(1+z^2) & 1 & 0 \\ -w_1(1+z^2) & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & -d_1(1+z^{-2}) - c_1(z^2+z^{-4}) & -n_1(1+z^{-2}) - m_1(z^2+z^{-4}) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} B_1(2\omega)B_0(2\omega) \begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$

$$\begin{bmatrix} \widetilde{p}(\omega) \\ \widetilde{q}^{(1)}(\omega) \\ \widetilde{q}^{(2)}(\omega) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & u_1(1+z^{-2}) & w_1(1+z^{-2}) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ d_1(1+z^2) + c_1(z^{-2}+z^4) & 1 & 0 \\ n_1(1+z^2) + m_1(z^{-2}+z^4) & 0 & 1 \end{bmatrix} \widetilde{B}_1(2\omega)\widetilde{B}_0(2\omega) \begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$

where B_0, B_1 and $\widetilde{B}_0, \widetilde{B}_1$ are defined by (47) and (48) respectively.

For the above frame filter banks, we can choose the parameters such that $\widetilde{\phi}$ is the C^4 5th degree B-spline. For example, if

$$t = 0, b = \frac{1}{2}, d = \frac{1}{4}, w = \frac{1}{2}, w_1 = 0, u = \frac{1}{2} - \frac{1}{4c_1}, d_1 = -c_1, m_1 = -\frac{3}{8} - n_1, u_1 = \frac{1}{4c_1},$$

then

$$\widetilde{p}(\omega) = \frac{1}{64} e^{i6\omega} (1 + e^{-4i\omega})^2 (1 + e^{-i\omega})^4, \tag{57}$$

$p(\omega)$ has sum rule rule order 2, $q^{(2)}(\omega)$ has vanishing moment order 4, and $q^{(1)}(\omega)$, $\widetilde{q}^{(1)}(\omega)$ and $\widetilde{q}^{(2)}(\omega)$ have vanishing moment order 2. Thus the corresponding $\widetilde{\phi}$ is the C^4 $5th$ degree B-spline supported on $[-6, 6]$. The resulting $p(\omega)$ depends on c_1, n_1. If we choose $c_1 = 1, n_1 = 0.17074863392459$, then the resulting ϕ is in $W^{1.00140}$; while if $c_1 = 1, n_1 = \frac{11}{64}$, then the resulting $p(\omega)$ has sum rule order 4 with ϕ in $W^{0.94455}$. In the following, we provide the resulting filters with $c_1 = 1, n_1 = \frac{11}{64}$ and other parameters given above:

$$
\begin{aligned}
p(\omega) &= \tfrac{1}{256} z^{-2}\{80 - 11(z + \tfrac{1}{z}) - 24(z^2 + \tfrac{1}{z^2}) + 3(z^3 + \tfrac{1}{z^3})\}(1+z)^4,\\
q^{(1)}(\omega) &= \tfrac{1}{1024}\{84 + 574(z + \tfrac{1}{z}) + 304(z^2 + \tfrac{1}{z^2}) + 101(z^3 + \tfrac{1}{z^3}) + 6(z^4 + \tfrac{1}{z^4}) - 3(z^5 + \tfrac{1}{z^5})\}(1-z)^2,\\
q^{(2)}(\omega) &= \tfrac{1}{4} z^{-1}(1-z)^4,\\
\widetilde{q}^{(1)}(\omega) &= \tfrac{1}{16} z^{-4}(1+z^2)(1+z)^6(1-z)^2,\\
\widetilde{q}^{(2)}(\omega) &= -\tfrac{1}{1024}\{1354 + 1054(z + \tfrac{1}{z}) + 584(z^2 + \tfrac{1}{z^2}) + 210(z^3 + \tfrac{1}{z^3}) + 35(z^4 + \tfrac{1}{z^4})\}(1-z)^2,
\end{aligned}
$$

where $z = e^{-i\omega}$, and $\widetilde{p}(\omega)$ is given by (57). The pictures of the corresponding scaling functions and framelets are shown in Fig. 14.

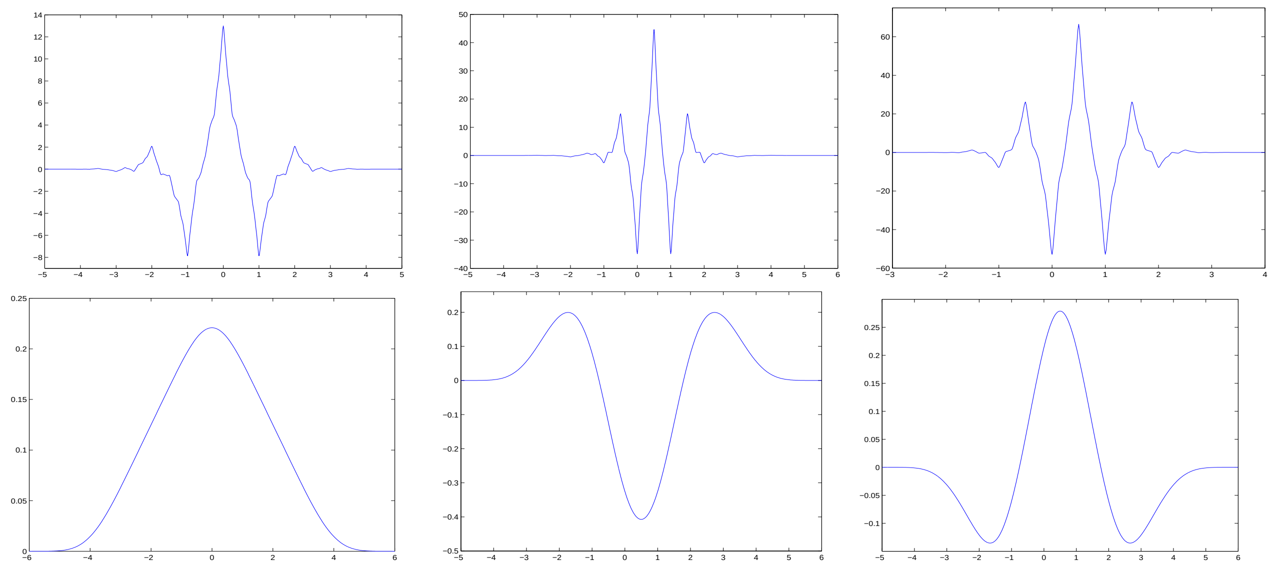

Figure 14: *Top (from left to right): $\phi, \psi^{(1)}, \psi^{(2)}$ with $\phi \in W^{1.00140}$; Bottom (from left to right): $\widetilde{\phi}, \widetilde{\psi}^{(1)}, \widetilde{\psi}^{(2)}$ with $\widetilde{\phi}$ being C^4 5th degree B-spline supported on [-6, 6]*

One may choose the parameters such that both $q^{(1)}(\omega)$ and $q^{(2)}(\omega)$ have vanishing moment order 4. For example, if

$$
\begin{aligned}
&d = \tfrac{1}{4},\ u_1 = 0,\ b = u,\ w = u(1-2w_1),\ c_1 = \tfrac{1}{4w_1} - \tfrac{1}{8uw_1} - d_1 - \tfrac{3}{8},\\
&n_1 = \tfrac{5}{32uw1} - \tfrac{5}{16w_1},\ m_1 = \tfrac{1}{16w_1} - \tfrac{1}{32uw_1},\ t = 1 - \tfrac{2}{3w_1} + \tfrac{1}{3uw_1},
\end{aligned}
$$

then the resulting $p(\omega)$ and $\widetilde{p}(\omega)$ have sum rule rule orders 2 and 4 respectively, $q^{(1)}(\omega)$ and $q^{(2)}(\omega)$ have vanishing moment order 4, and $\widetilde{q}^{(1)}(\omega)$ and $\widetilde{q}^{(2)}(\omega)$ have vanishing moment order 2. We can choose the parameters such that the resulting $\widetilde{\phi}$ is in C^3 with $\phi \in L^2(\mathbb{R}^2)$. For example, if

$$
[w_1,\ d_1,\ u] = [\frac{39}{64},\ -\frac{35}{64},\ \frac{59}{128}],
$$

then ϕ and $\widetilde{\phi}$ are in $W^{0.01083}$ and $W^{3.52895}$ respectively. We can choose the parameters such that ϕ is smoother. For example, if

$$[w_1,\ d_1,\ u] = [\frac{63}{128},\ -\frac{21}{128},\ \frac{15}{16}],$$

then ϕ and $\widetilde{\phi}$ are in $W^{1.14749}$ and $W^{2.50565}$ respectively.

The biorthogonal lowpass filters ${}_4p(\omega), {}_4\widetilde{p}(\omega)$ in Example 2 have the same supports as those of the frame lowpass filters $p(\omega), \widetilde{p}(\omega)$ for the 4-step frame multiresolution algorithms considered above. Observe that the frame synthesis scaling function $\widetilde{\phi}$ could be C^4 B-spline with the frame analysis scaling function ϕ having certain smoothness. This cannot happen for the biorthogonal scaling functions if they have the same supports as the frame scaling functions. Thus, compared with the biorthogonal system, the frame system does provide certain flexibility for construction.

In general, the frame filter banks corresponding to algorithms with more steps can be given as above. More precisely, with $z = e^{-i\omega}$, let $C(\omega)$ be the matrix of the form

$$C(\omega) = \begin{bmatrix} 1 & L_1(z) & L_2(z) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{58}$$

where $L_1(z)$ and $L_2(z)$ are Laurent polynomials satisfying $L_1(\frac{1}{z}) = zL_1(z), L_2(\frac{1}{z}) = zL_2(z)$, namely they are Laurent polynomials of the form:

$$L(z) = l_1(1 + \frac{1}{z}) + l_2(z + \frac{1}{z^2}) + \cdots + l_m(z^{m-1} + \frac{1}{z^m}), \tag{59}$$

for some positive integer m and real numbers l_j. Clearly, $\widetilde{C}(\omega) = (C(\omega)^{-1})^*$ is

$$\widetilde{C}(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ -L_1(\frac{1}{z}) & 1 & 0 \\ -L_2(\frac{1}{z}) & 0 & 1 \end{bmatrix}.$$

Then the frame filter banks corresponding to algorithms with K $(K \geq 1)$ steps can be given as

$$[p(\omega),\ q^{(1)}(\omega),\ q^{(2)}(\omega)]^T = B_{K-1}(2\omega)B_{K-2}(2\omega)\cdots B_1(2\omega)B_0(2\omega)\left[1,\ e^{-i\omega}\right]^T, \tag{60}$$

$$[\widetilde{p}(\omega),\ \widetilde{q}^{(1)}(\omega),\ \widetilde{q}^{(2)}(\omega)]^T = \frac{1}{2}\widetilde{B}_{K-1}(2\omega)\widetilde{B}_{K-2}(2\omega)\cdots \widetilde{B}_1(2\omega)\widetilde{B}_0(2\omega)\left[1,\ e^{-i\omega}\right]^T, \tag{61}$$

where B_0 and $\widetilde{B}_0$ are defined by (47) and (48), each $B_k(\omega)$, $1 \leq k \leq K-1$, is a matrix $C(\omega)$ of the form (58) or $\widetilde{C}(\omega)$, and $\widetilde{B}_k(\omega) = (B_k(\omega)^{-1})^*$. The next proposition shows that the framelets obtained by these algorithms have the uniform symmetry.

Proposition 1. *Let $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ be the biorthogonal frame filter banks defined by (60) and (61). Then*

$$p(-\omega) = p(\omega),\ q^{(\ell)}(-\omega) = e^{i2\omega}q^{(\ell)}(\omega),\ \widetilde{p}(-\omega) = \widetilde{p}(\omega),\ \widetilde{q}^{(\ell)}(-\omega) = e^{i2\omega}\widetilde{q}^{(\ell)}(\omega),\ \ell = 1, 2.$$

Furthermore, the associated scaling functions $\phi, \widetilde{\phi}$, and framelets $\psi^{(\ell)}, \widetilde{\psi}^{(\ell)}, \ell = 1, 2$ satisfy

$$\phi(x) = \phi(-x),\ \psi^{(\ell)}(x) = \psi^{(\ell)}(1-x),\ \widetilde{\phi}(x) = \widetilde{\phi}(-x),\ \widetilde{\psi}^{(\ell)}(x) = \widetilde{\psi}^{(\ell)}(1-x).$$

Proof. One can easily verify that

$$B_k(-\omega) = \mathrm{diag}(1, e^{i\omega}, e^{i\omega})B_k(\omega)\mathrm{diag}(1, e^{-i\omega}, e^{-i\omega}),\ B_0(-\omega) = \mathrm{diag}(1, e^{i\omega}, e^{i\omega})B_0(\omega)\mathrm{diag}(1, e^{-i\omega}), \tag{62}$$

where $1 \le k \le K-1$, which implies

$$[p(-\omega), q^{(1)}(-\omega), q^{(2)}(-\omega)]^T = \mathrm{diag}(1, e^{i2\omega}, e^{i2\omega})[p(\omega), q^{(1)}(\omega), q^{(2)}(\omega)]^T,$$

as desired. The symmetry of $\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}$ follows from the fact that $\widetilde{B}_k(\omega)$ and $\widetilde{B}_0(\omega)$ also satisfy (62).

From $\hat{\phi}(\omega) = \Pi_{j=1}^{\infty} p(2^{-j}\omega)\hat{\phi}(0)$ and $p(-\omega) = p(\omega)$, we have

$$\hat{\phi}(-\omega) = \Pi_{j=1}^{\infty} p(-2^{-j}\omega)\hat{\phi}(0) = \Pi_{j=1}^{\infty} p(2^{-j}\omega)\hat{\phi}(0) = \hat{\phi}(\omega).$$

Thus $\phi(-x) = \phi(x)$.

From $\hat{\psi}^{(\ell)}(\omega) = q^{(\ell)}(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2})$ and $q^{(\ell)}(-\omega) = e^{i2\omega}q^{(\ell)}(\omega)$, we have

$$\hat{\psi}^{(\ell)}(-\omega) = q^{(\ell)}(-\frac{\omega}{2})\hat{\phi}(-\frac{\omega}{2}) = e^{i\omega}q^{(\ell)}(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2}) = e^{i\omega}\hat{\psi}^{(\ell)}(\omega).$$

Thus $\psi^{(\ell)}(-x) = \psi^{(\ell)}(x+1)$, as desired. The proof for the symmetry of $\widetilde{\phi}$ and $\widetilde{\psi}^{(\ell)}$ is similar. $\diamondsuit$

4 Bi-frames with uniform symmetry: Type II

For a pair of biorthogonal frame filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$, let $\widetilde{c}_k$ and $d_k^{(1)}, d_k^{(2)}$ be the lowpass output and highpass outputs of input c_k defined by (2) with the analysis frame filter bank $\{p, q^{(1)}, q^{(2)}\}$. As in §3, denote $\widetilde{v}_k = \widetilde{c}_k$, $\widetilde{f}_k = d_k^{(1)}$, $\widetilde{e}_k = d_k^{(2)}$. In §3, we associate both $\widetilde{f}_k$ and $\widetilde{e}_k$ with the odd node $2k+1$. In this section we consider the frame algorithms, called type II frame algorithms, by associating $\widetilde{f}_k$ with the even node $2k$ and $\widetilde{e}_k$ with the odd node $2k+1$. We find that compared with type I frame algorithms, type II frame algorithms yield smoother frames and analysis highpass filters with higher vanishing moment orders. Type II frame algorithms with 3 and 4 step iterations, and the 4-point interpolatory subdivision scheme-based bi-frames are studied in the following 3 subsections respectively.

4.1 3-step type II frame algorithm

In this subsection we consider a 3-step type II frame algorithm. The decomposition algorithm is given in (63)-(65) and shown in Fig. 15, where b, d, n, u, w, d_1, n_1 are some constants. Namely, first we obtain v'' and f'' associated with even nodes $2k$ by the formulas in (63). After that, with the obtained v'', f'', we obtain one highpass output $\widetilde{e}$ that is associated with odd nodes $2k+1$ and given by (64). Finally, with the obtained $\widetilde{e}$, we replace v'' and f'' by $\widetilde{v}$ and $\widetilde{f}$ given in (65). This step gives the lowpass output $\widetilde{v}$ and the other highpass output $\widetilde{f}$ associated with even nodes.

3-step Type II Frame Decomposition Algorithm:

Step 1. $v'' = \frac{1}{b}\{v - d(e_{-1} + e_0)\},\ f'' = v - n(e_{-1} + e_0);$ (63)

Step 2. $\widetilde{e} = e - u(v''_0 + v''_1) - w(f''_0 + f''_1);$ (64)

Step 3. $\widetilde{v} = v'' - d_1(\widetilde{e}_{-1} + \widetilde{e}_0),\ \widetilde{f} = f'' - n_1(\widetilde{e}_{-1} + \widetilde{e}_0).$ (65)

3-step Type II Frame Reconstruction Algorithm:

Step 1. $v'' = \widetilde{v} + d_1(\widetilde{e}_{-1} + \widetilde{e}_0),\ f'' = \widetilde{f} + n_1(\widetilde{e}_{-1} + \widetilde{e}_0);$ (66)

Step 2. $e = \widetilde{e} + u(v_0'' + v_1'') + w(f_0'' + f_1'');$ (67)

Step 3. $v = t\{bv'' + d(e_{-1} + e_0)\} + (1-t)\{f'' + n(e_{-1} + e_0)\}.$ (68)

The reconstruction algorithm is given in (66)-(68) and shown in Fig. 16, where b, d, n, u, w, d_1, n_1

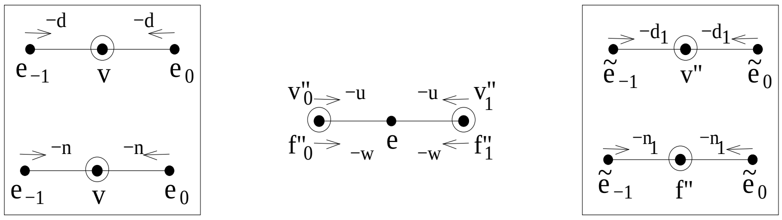

Figure 15: *Left: Decomposition Step 1; Middle: Decomposition Step 2; Right: Decomposition Step 3*

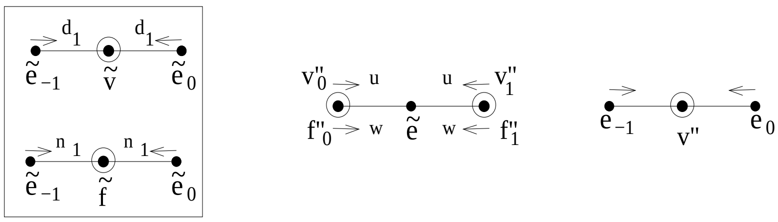

Figure 16: *Left: Reconstruction Step 1; Middle: Reconstruction Step 2; Right: Reconstruction Step 3*

are the same constants as in the decomposition algorithm and $t \in \mathbb{R}$. That is, first we obtain v'', f'' associated with even nodes $2k$ by the formulas in (66). After that, with the obtained v'', f'', we replace $\widetilde{e}$ of the highpass output associated with odd nodes by e given in (67). This step recovers original data c_{2k+1} associated with odd nodes $2k+1$. Finally, we obtain v from (68). This step recovers original data c_{2k} associated with even nodes.

As in §2, one can obtain that the filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ corresponding to the frame algorithm (63)-(68) are

$$\left[p(\omega),\ q^{(1)}(\omega),\ q^{(2)}(\omega)\right]^T = D_2(2\omega)D_1(2\omega)D_0(2\omega)\left[1,\ e^{-i\omega}\right]^T, \tag{69}$$

$$\left[\widetilde{p}(\omega),\ \widetilde{q}^{(1)}(\omega),\ \widetilde{q}^{(2)}(\omega)\right]^T = \frac{1}{2}\widetilde{D}_2(2\omega)\widetilde{D}_1(2\omega)\widetilde{D}_0(2\omega)\left[1,\ e^{-i\omega}\right]^T, \tag{70}$$

where, with $z = e^{-i\omega}$,

$$\begin{cases} D_2(\omega) = \begin{bmatrix} 1 & 0 & -d_1(1+\frac{1}{z}) \\ 0 & 1 & -n_1(1+\frac{1}{z}) \\ 0 & 0 & 1 \end{bmatrix}, D_1(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -u(1+z) & -w(1+z) & 1 \end{bmatrix}, \\ D_0(\omega) = \begin{bmatrix} \frac{1}{b} & -\frac{d}{b}(1+\frac{1}{z}) \\ 1 & -n(1+\frac{1}{z}) \\ 0 & 1 \end{bmatrix}, \end{cases} \tag{71}$$

$$\begin{cases} \widetilde{D}_2(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ d_1(1+z) & n_1(1+z) & 1 \end{bmatrix}, \widetilde{D}_1(\omega) = \begin{bmatrix} 1 & 0 & u(1+\frac{1}{z}) \\ 0 & 1 & w(1+\frac{1}{z}) \\ 0 & 0 & 1 \end{bmatrix}, \\ \widetilde{D}_0(\omega) = \begin{bmatrix} tb & 0 \\ 1-t & 0 \\ (td+(1-t)n)(1+z) & 1 \end{bmatrix}. \end{cases} \tag{72}$$

After solving the system of equations for sum rule order 4 of $\widetilde{p}$, sum rule order 2 of p and for vanishing moment order 2 of $q^{(1)}, q^{(2)}$ and $\widetilde{q}^{(1)}, \widetilde{q}^{(2)}$, we have

$$n = \frac{1}{2},\ u = \frac{1}{2},\ d_1 = -\frac{3}{8},\ d = \frac{1}{2} - \frac{b}{2},\ w = \frac{1}{6b} - \frac{1}{3},\ t = \frac{1}{2b}.$$

The resulting filter $\widetilde{p}$ is

$$\widetilde{p}(\omega) = \frac{1}{16} e^{i2\omega}(1 + e^{-i\omega})^4, \tag{73}$$

and $p(\omega)$ depends on parameter b. Furthermore, if

$$n_1 = \frac{3b}{8(2b-1)},$$

then $q^{(1)}(\omega)$ has vanishing moment order 4. Thus the corresponding $\widetilde{\phi}$ is the C^2 cubic B-spline supported on $[-2, 2]$. For $p(\omega)$, if $b = \frac{4}{3}$, then $p(\omega)$ has sum rule order 4 with corresponding $\phi \in W^{1.82037}$. If we choose $b = 1.33693417502911$, then $\phi \in W^{1.87040}$; and if $b = \frac{345}{256}$, then $\phi \in W^{1.86992}$. With all these choices of b, the resulting $p(\omega)$ is supported on $[-3, 3]$. Recall from Example 1 that if the synthesis scaling function $\widetilde{\phi}$ is the C^2 cubic B-spline with its corresponding lowpass filter $\widetilde{p}$ given by (73), then its analysis scaling function ϕ supported on [-4, 4] has a low smooth order. However, for the bi-frame system, we can construct the analysis scaling function ϕ such that ϕ is in C^1 and it has a smaller support [-3, 3]. Furthermore, the corresponding analysis lowpass filter $p(\omega)$ has sum rule order 4. In the following we provide all the selected numbers with $b = \frac{4}{3}$:

$$[b,\ d,\ n,\ u,\ w,\ d_1,\ n_1,\ t] = [\frac{4}{3},\ -\frac{1}{6},\ \frac{1}{2},\ \frac{1}{2},\ -\frac{5}{24},\ -\frac{3}{8},\ \frac{3}{10},\ \frac{3}{8}]. \tag{74}$$

The filters corresponding to these selected parameters are

$$\begin{aligned} p(\omega) &= \tfrac{1}{16} e^{2i\omega}(3 - e^{-i\omega} - e^{i\omega})(1 + e^{-i\omega})^4, \\ q^{(1)}(\omega) &= \tfrac{1}{20} e^{2i\omega}(5 + e^{-i\omega} + e^{i\omega})(1 - e^{-i\omega})^4, \\ q^{(2)}(\omega) &= -\tfrac{1}{6}(3 + e^{-i\omega} + e^{i\omega})(1 - e^{-i\omega})^2, \\ \widetilde{q}^{(1)}(\omega) &= -\tfrac{5}{192} e^{i\omega}(6 + e^{-i\omega} + e^{i\omega})(1 - e^{-i\omega})^2, \\ \widetilde{q}^{(2)}(\omega) &= -\tfrac{1}{32} e^{i\omega}(4 + e^{-i\omega} + e^{i\omega})(1 + e^{-i\omega})^2(1 - e^{-i\omega})^2, \end{aligned}$$

and $\widetilde{p}(\omega)$ is given by (73). The pictures of the corresponding scaling functions and framelets are shown in Fig. 17.

Before we move on to the next subsection, we remark here on the algorithm with 2 steps. When $d_1 = n_1 = 0$, then the 3-step algorithm is reduced to a 2-step algorithm. More precisely, the 2-step decomposition algorithm is given by (63)-(64) (with $\widetilde{v} = v'', \widetilde{f} = f''$) and the 2-step reconstruction algorithm is given by (67)-(68) (with $v'' = \widetilde{v}, f'' = \widetilde{f}$). The corresponding filter banks, also denoted by $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$, are given by (69) and (70) with

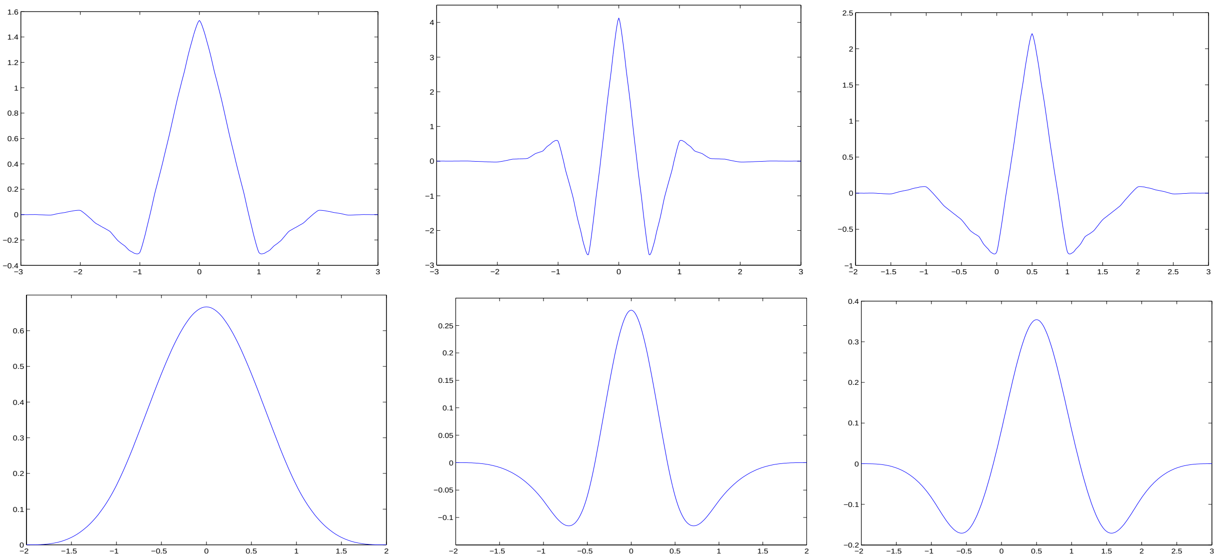

Figure 17: *Top (from left to right):* $\phi, \psi^{(1)}, \psi^{(2)}$ *with* $\phi \in W^{1.82037}$*; Bottom (from left to right):* $\widetilde{\phi}, \widetilde{\psi}^{(1)}, \widetilde{\psi}^{(2)}$ *with* $\widetilde{\phi}$ *being* C^2 *cubic B-spline supported on [-2, 2]*

$D_2(\omega) = \widetilde{D}_2(\omega) = I_3,\ D_1(\omega), D_0(\omega)$ and $\widetilde{D}_1(\omega), \widetilde{D}_0(\omega)$ are defined by (71) and (72) respectively. If we drop the condition for vanishing moments of synthesis highpass filters, we can get $\phi, \widetilde{\phi}$ both to be B-splines. More precisely, after solving the system of equations for sum rule order 4 of $\widetilde{p}$, sum rule order 2 of p and for vanishing moment order 2 of both $q^{(1)}$ and $q^{(2)}$, we have

$$b = 2,\ d = -\frac{1}{2},\ n = \frac{1}{2},\ u = \frac{1}{2},\ t = \frac{1}{4}. \tag{75}$$

The resulting filters are

$$\begin{array}{l} p(\omega) = \frac{1}{4}e^{i\omega}(1+e^{-i\omega})^2,\ q^{(1)}(\omega) = -\frac{1}{2}e^{i\omega}(1-e^{-i\omega})^2, \\ \widetilde{p}(\omega) = \frac{1}{16}e^{i2\omega}(1+e^{-i\omega})^4,\ \widetilde{q}^{(2)}(\omega) = \frac{1}{8}(1+4e^{-i\omega}+e^{-i2\omega}), \end{array}$$

and $q^{(2)}(\omega), \widetilde{q}^{(1)}(\omega)$ depend on parameter w. If $w = 0$, then

$$q^{(2)}(\omega) = -\frac{1}{8}(4+e^{i\omega}+e^{-i\omega})(1-e^{-i\omega})^2,\ \widetilde{q}^{(1)}(\omega) = \frac{3}{8};$$

while for $w = \frac{3}{4}$, the resulting $q^{(2)}(\omega), \widetilde{q}^{(1)}(\omega)$ are

$$q^{(2)}(\omega) = \frac{1}{4}e^{i2\omega}(1-e^{-i\omega})^4,\ \widetilde{q}^{(1)}(\omega) = \frac{3}{32}e^{i2\omega}(1+e^{-i\omega})^4.$$

Thus for any w, the resulting ϕ is the continuous linear B-spline supported on $[-1, 1]$ and $\widetilde{\phi}$ is the C^2 cubic B-spline supported on $[-2, 2]$. When $w = 0$, $\widetilde{\psi}^{(1)}(x) = \frac{3}{4}\widetilde{\phi}(2x)$; and for $w = \frac{3}{4}$, $q^{(2)}(\omega)$ has vanishing moments of order 4 and $\widetilde{q}^{(1)}(\omega) = \frac{3}{2}\widetilde{p}(\omega)$. Pictures of the scaling functions and framelets corresponding to $w = 0$ are shown in Fig. 18.

If we choose

$$b = 2,\ d = -\frac{1}{2},\ n = \frac{1}{2},\ u = \frac{1}{2},\ w = \frac{1}{4},\ t = \frac{1}{2},$$

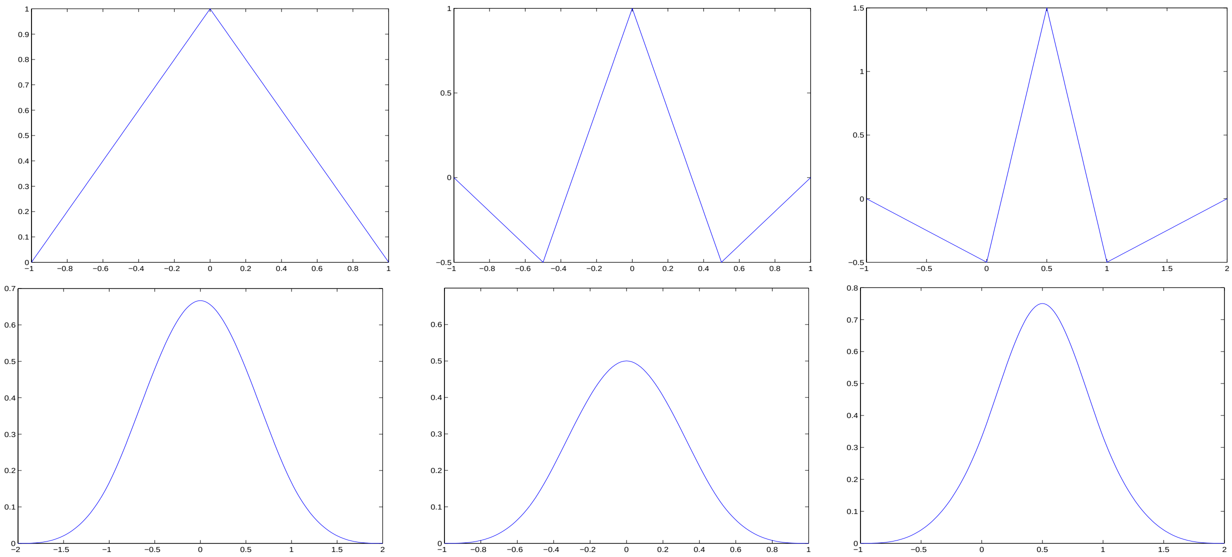

Figure 18: *Top (from left to right): $\phi, \psi^{(1)}, \psi^{(2)}$ with ϕ being continuous linear B-spline supported on [-1, 1]; Bottom (from left to right): $\widetilde{\phi}, \widetilde{\psi}^{(1)}, \widetilde{\psi}^{(2)}$ with $\widetilde{\phi}$ being C^2 cubic B-spline supported on [-2, 2]*

then both ϕ and $\widetilde{\phi}$ are the continuous linear B-spline supported on [-1, 1] and the resulting filters are

$$\begin{array}{l} p(\omega) = \widetilde{p}(\omega) = \frac{1}{4}e^{i\omega}(1+e^{-i\omega})^2,\ q^{(1)}(\omega) = -\frac{1}{2}e^{i\omega}(1-e^{-i\omega})^2, \\ q^{(2)}(\omega) = -\frac{1}{2}(1-e^{-i\omega})^2,\ \widetilde{q}^{(1)}(\omega) = \frac{1}{8}e^{i\omega}(1+e^{-i\omega})^2,\ \widetilde{q}^{(2)}(\omega) = \frac{1}{2}e^{-i\omega}. \end{array}$$

Similarly, for the 3-step algorithm, if we drop the condition for vanishing moments of synthesis highpass filters, we can also get $\phi, \widetilde{\phi}$ both to be B-splines. For example, if we choose

$$b = 8,\ d = -\frac{7}{2},\ n = \frac{1}{2},\ u = \frac{1}{2},\ w = \frac{5}{16},\ d_1 = \frac{1}{4}, n_1 = \frac{2}{5},\ t = \frac{1}{16},$$

then we have

$$\begin{array}{l} p(\omega) = \frac{1}{64}e^{i3\omega}(1+e^{-i\omega})^6,\ q^{(1)}(\omega) = \frac{1}{40}e^{i2\omega}(10+e^{i\omega}+e^{-i\omega})(1-e^{-i\omega})^4, \\ q^{(2)}(\omega) = -\frac{1}{16}(8+e^{i\omega}+e^{-i\omega})(1-e^{-i\omega})^2, \\ \widetilde{p}(\omega) = \frac{1}{16}e^{i2\omega}(1+e^{-i\omega})^4,\ \widetilde{q}^{(1)}(\omega) = \frac{5}{128}\{14+4(e^{i\omega}+e^{-i\omega})+e^{i2\omega}+e^{-i2\omega}\}, \\ \widetilde{q}^{(2)}(\omega) = \frac{1}{32}\{10+2(e^{i\omega}+e^{-i\omega})+e^{i2\omega}+e^{-i2\omega}\}(1+e^{-i\omega})^2. \end{array}$$

Thus the resulting ϕ is the C^3 quartic B-spline supported on $[-3, 3]$ and $\widetilde{\phi}$ is the C^2 cubic spline supported on $[-2, 2]$.

The 1-D frame algorithms can be used as boundary algorithms for boundary vertices on open surfaces for surface multiresolution processing. These frame algorithms can also be used for curve processing. Here, as an example, we use the above 2-step algorithm (with $w = \frac{3}{4}$ and other parameters given by (75)) for curve noise-removing. The left column of Fig. 19 are the original curves. The curves with white noise are shown in the middle column of Fig. 19. We show the denoised curves in the right column after we apply several times the 2-step algorithm and hard thresholding process for denoising.

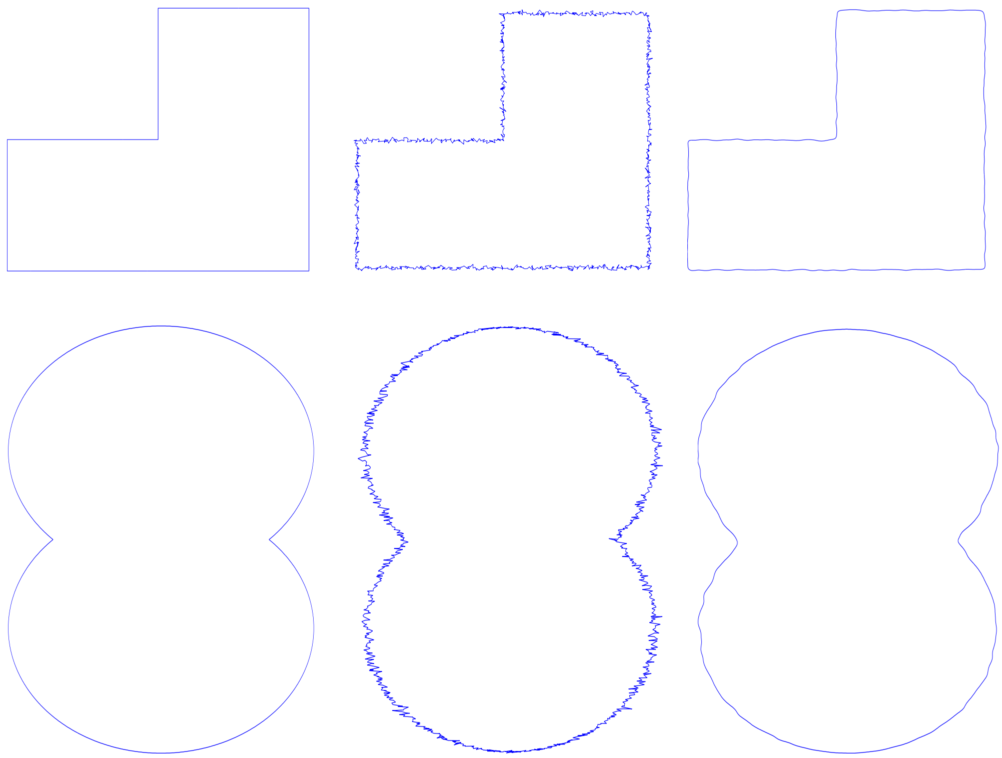

Figure 19: *Left column: Original curves; Middle column: Noised curves; Right: Denoised curves*

4.2 4-step type II frame algorithm

In this subsection, we consider a 4-step type II frame algorithm. The decomposition algorithm is given in (76)-(79) and shown in Fig. 20, and the reconstruction algorithm is given in (80)-(83) and shown in Fig. 21, where $b, d, n, u, w, d_1, n_1, u_1, w_1, t$ are some constants.

4-step Type II Frame Decomposition Algorithm:

$$\text{Step 1. } v'' = \frac{1}{b}\{v - d(e_{-1} + e_0)\}, \; f'' = v - n(e_{-1} + e_0); \tag{76}$$

$$\text{Step 2. } e'' = e - u(v''_0 + v''_1) - w(f''_0 + f''_1); \tag{77}$$

$$\text{Step 3. } \widetilde{v} = v'' - d_1(e''_{-1} + e''_0), \; \widetilde{f} = f'' - n_1(e''_{-1} + e''_0); \tag{78}$$

$$\text{Step 4. } \widetilde{e} = e'' - u_1(\widetilde{v}_0 + \widetilde{v}_1) - w_1(\widetilde{f}_0 + \widetilde{f}_1). \tag{79}$$

4-step Type II Frame Reconstruction Algorithm:

$$\text{Step 1. } e'' = \widetilde{e} + u_1(\widetilde{v}_0 + \widetilde{v}_1) + w_1(\widetilde{f}_0 + \widetilde{f}_1); \tag{80}$$

$$\text{Step 2. } v'' = \widetilde{v} + d_1(e''_{-1} + e''_0), \; f'' = \widetilde{f} + n_1(e''_{-1} + e''_0); \tag{81}$$

$$\text{Step 3. } e = e'' + u(v''_0 + v''_1) + w(f''_0 + f''_1); \tag{82}$$

$$\text{Step 4. } v = t\{bv'' + d(e_{-1} + e_0)\} + (1-t)\{f'' + n(e_{-1} + e_0)\}. \tag{83}$$

With careful calculations, one can obtain that the filter banks $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$

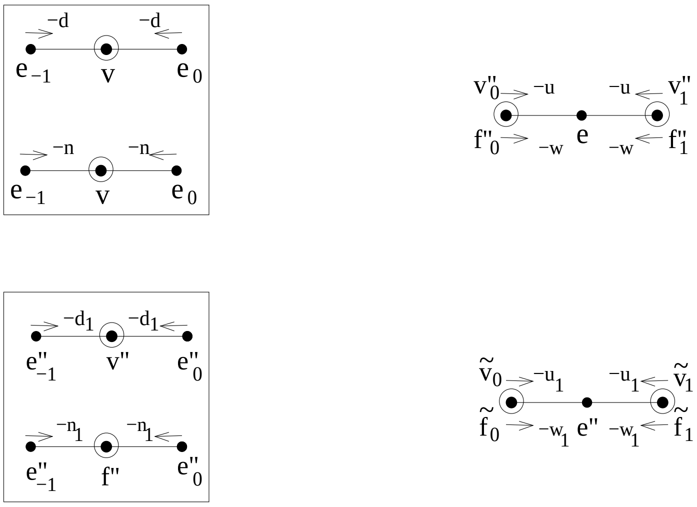

Figure 20: *Top-left: Decomposition Step 1; Top-right: Decomposition Step 2; Bottom-left: Decomposition Step 3; Bottom-right: Decomposition Step 4*

corresponding to the frame algorithm (76)-(83) are

$$\begin{bmatrix} p(\omega) \\ q^{(1)}(\omega) \\ q^{(2)}(\omega) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -u_1(1+e^{-i2\omega}) & -w_1(1+e^{-i2\omega}) & 1 \end{bmatrix} D_2(2\omega)D_1(2\omega)D_0(2\omega) \begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$

$$\begin{bmatrix} \widetilde{p}(\omega) \\ \widetilde{q}^{(1)}(\omega) \\ \widetilde{q}^{(2)}(\omega) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 0 & u_1(1+e^{i2\omega}) \\ 0 & 1 & w_1(1+e^{i2\omega}) \\ 0 & 0 & 1 \end{bmatrix} \widetilde{D}_2(2\omega)\widetilde{D}_1(2\omega)\widetilde{D}_0(2\omega) \begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix},$$

where $D_2(\omega), D_1(\omega), D_0(\omega)$ and $\widetilde{D}_2(\omega), \widetilde{D}_1(\omega), \widetilde{D}_0(\omega)$ are defined by (71) and (72) respectively.

Solving the system of equations for sum rule order 2 of p, sum rule order 8 of $\widetilde{p}$ and for vanishing moment order 2 of $q^{(1)}, q^{(2)}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}$, we have

$$u = \tfrac{3}{4},\ u_1 = -\tfrac{1}{2},\ d = \tfrac{1}{2} - \tfrac{13}{16}b,\ n = \tfrac{8-29b}{16(1-8b)},\ w = \tfrac{1-8b}{20b},\ d_1 = -\tfrac{5}{16},\ n_1 = \tfrac{35b}{16(1-8b)},\ t = \tfrac{1}{8b}.$$

The resulting filter $\widetilde{p}$ is

$$\widetilde{p}(\omega) = \frac{1}{256} e^{i4\omega}(1+e^{-i\omega})^8. \tag{84}$$

Thus the corresponding $\widetilde{\phi}$ is the C^6 7th degree B-spline supported on $[-4, 4]$. The resulting $p(\omega)$ depends on parameter b. Furthermore, if

$$w_1 = \frac{4}{5} - \frac{1}{10b},$$

then $q^{(1)}(\omega)$ has vanishing moment order 4. When $b = 1$, $p(\omega)$ has sum rule order 4 with the corresponding $\phi \in W^{1.38583}$. If we choose $b = \frac{31}{30}$, then the corresponding ϕ is in $W^{1.53528}$. Therefore, the resulting ϕ is in C^1. In the following we provide all the selected numbers with $b = \frac{31}{30}$:

$$[b,\ d,\ n,\ u,\ w,\ d_1,\ n_1, u_1, w_1] = [\frac{31}{30}, -\frac{163}{480}, \frac{659}{3488}, \frac{3}{4}, -\frac{109}{310}, -\frac{5}{16}, -\frac{1085}{3488}, -\frac{1}{2}, \frac{109}{155}].$$

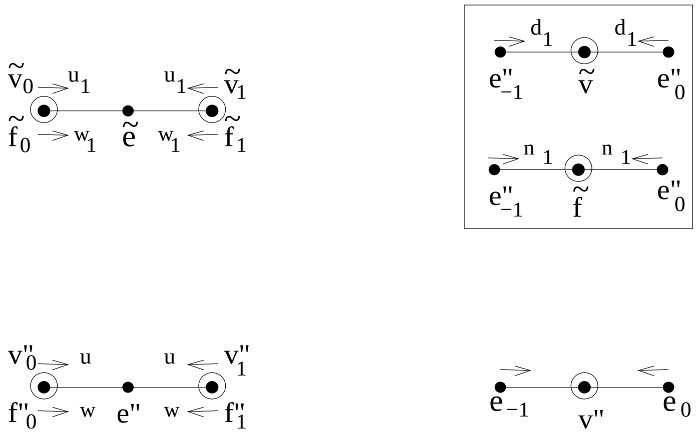

Figure 21: *Top-left: Reconstruction Step 1; Top-right: Reconstruction Step 2; Bottom-left: Reconstruction Step 3; Bottom-right: Reconstruction Step 4*

The pictures of the scaling functions and framelets corresponding to these selected parameters are shown in Fig. 22. The resulting $p(\omega), q^{(1)}(\omega), q^{(2)}(\omega), \widetilde{q}^{(1)}(\omega)$ and $\widetilde{q}^{(2)}(\omega)$ are supported on [-3, 3], [-3, 3], [-3, 5], [-4, 4] and [-2, 4] respectively.

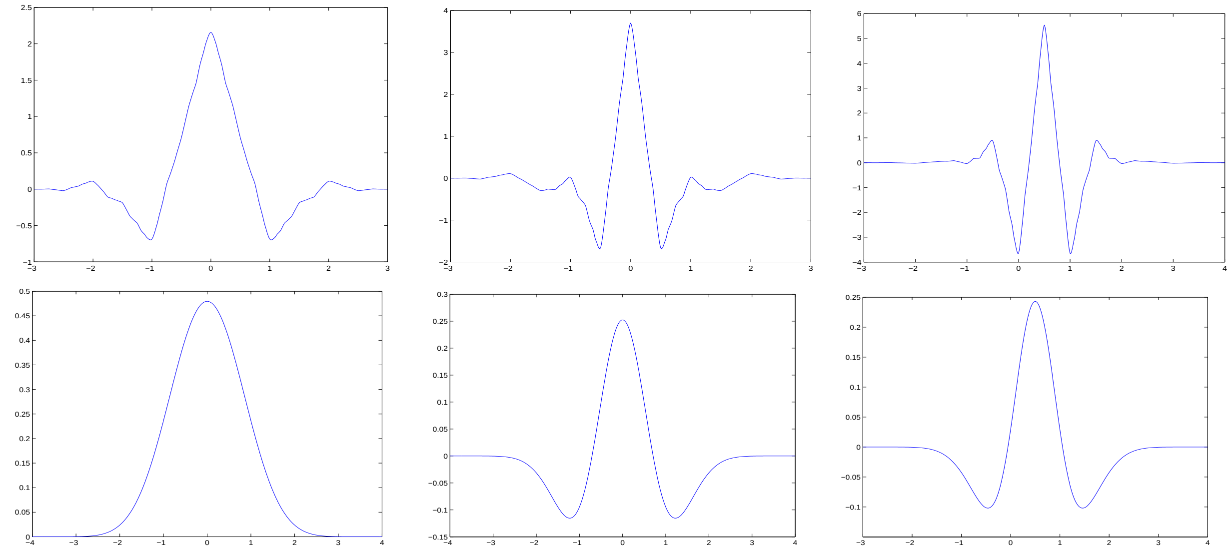

Figure 22: *Top (from left to right):* $\phi, \psi^{(1)}, \psi^{(2)}$ *with* $\phi \in W^{1.53528}$*; Bottom (from left to right):* $\widetilde{\phi}, \widetilde{\psi}^{(1)}, \widetilde{\psi}^{(2)}$ *with* $\widetilde{\phi}$ *being* C^6 *7th degree B-spline supported on [-4, 4]*

4.3 4-point interpolatory subdivision scheme-based bi-frames and general case

In this subsection, first we provide bi-frames with the synthesis lowpass filter being the symbol of the 4-point interpolatory scheme in [23]. After that we consider the general case for type II framelets.

The decomposition algorithm of the 4-point interpolatory scheme-based bi-frames is the same as that of the 3-step Type II frame decomposition algorithm (63) -(65) except that Step 2 in (64) is replaced by Step 2′ in (85), and the reconstruction is the same as that of the 3-step Type II frame reconstruction algorithm (66)-(68) except that (67) is replaced by Step 2′ in (86). Refer to Fig. 23 for Step 2′ of decomposition and reconstruction algorithms.

Step 2 of 4-point Interpolatory Scheme-based Frame Decomposition Algorithm:

Step 2′. $\widetilde{e} = e - u(v_0'' + v_1'') - r(v_2'' + v_3'') - w(f_0'' + f_1'') - s(f_2'' + f_3'')$; (85)

Step 2 of 4-point Interpolatory Scheme-based Frame Reconstruction Algorithm:

Step 2′. $e = \widetilde{e} + u(v_0'' + v_1'') + r(v_2'' + v_3'') + w(f_0'' + f_1'') + s(f_2'' + f_3'')$. (86)

One can obtain that the corresponding filters are given by (69) and (70), where $D_2(\omega), D_0(\omega)$

Figure 23: *Left: Decomposition Step* 2′*; Right: Reconstruction Step* 2′

and $\widetilde{D}_2(\omega), \widetilde{D}_0(\omega)$ are given by (71) and (72) respectively, and $D_1(\omega)$ and $\widetilde{D}_1(\omega)$ are defined by

$$D_1(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -u(1+e^{-i\omega}) - r(e^{i\omega} + e^{-i2\omega}) & -w(1+e^{-i\omega}) - s(e^{i\omega} + e^{-i2\omega}) & 1 \end{bmatrix},$$

$$\widetilde{D}_1(\omega) = \begin{bmatrix} 1 & 0 & u(1+e^{i\omega}) + r(e^{-i\omega} + e^{i2\omega}) \\ 0 & 1 & w(1+e^{i\omega}) + s(e^{-i\omega} + e^{i2\omega}) \\ 0 & 0 & 1 \end{bmatrix}.$$

If $t = \frac{1}{b}$, $d = (1-b)n$, then the subdivision scheme derived from the resulting $\widetilde{p}(\omega)$ is interpolatory. Furthermore, if $r = -\frac{1}{16}$, $u = \frac{9}{16}$, then the scheme is the 4-point interpolatory scheme in [23] with $\widetilde{\phi} \in W^{2.44076}$. In addition, if

$$b = \frac{3}{2(1-8s)},\ w = \frac{1}{2b} - \frac{1}{2} - s,\ n = \frac{1}{2},\ d_1 = -\frac{1}{4},\ n_1 = \frac{3}{4(1+16s)},$$

then the resulting $p(\omega)$ has sum rule order 4, $q^{(2)}(\omega), \widetilde{q}^{(1)}(\omega), \widetilde{q}^{(2)}(\omega)$ have vanishing moment order 2, and $q^{(1)}(\omega)$ has vanishing moment order 4. We can choose s such that $p(\omega)$ has quite nice smoothness. For example, if $s = 0.02972961220002$, then the resulting ϕ is in $W^{3.30274}$, and if $s = \frac{1}{32}$, then $\phi \in W^{3.28254}$. In the latter case, the resulting $\widetilde{q}^{(1)}(\omega)$ has vanishing moment order 4. In the following, we provide the corresponding filters with $s = \frac{1}{32}$ and other parameters given above (with $z = e^{-i\omega}$):

$$\begin{aligned}
p(\omega) &= \tfrac{1}{2}(1 + \tfrac{19}{32}(z + \tfrac{1}{z}) - \tfrac{7}{64}(z^3 + \tfrac{1}{z^3}) + \tfrac{1}{64}(z^5 + \tfrac{1}{z^5})),\\
q^{(1)}(\omega) &= 1 - \tfrac{19}{32}(z + \tfrac{1}{z}) + \tfrac{7}{64}(z^3 + \tfrac{1}{z^3}) - \tfrac{1}{64}(z^5 + \tfrac{1}{z^5}),\\
q^{(2)}(\omega) &= \tfrac{1}{4}(\tfrac{7}{4}z - (z^3 + \tfrac{1}{z}) + \tfrac{1}{8}(z^5 + \tfrac{1}{z^3})),\\
\widetilde{p}(\omega) &= \tfrac{1}{2}(1 + \tfrac{9}{16}(z + \tfrac{1}{z}) - \tfrac{1}{16}(z^3 + \tfrac{1}{z^3})),\\
\widetilde{q}^{(1)}(\omega) &= \tfrac{1}{4}(1 - \tfrac{9}{16}(z + \tfrac{1}{z}) + \tfrac{1}{16}(z^3 + \tfrac{1}{z^3})),\\
\widetilde{q}^{(2)}(\omega) &= \tfrac{1}{8}(\tfrac{7}{4}z - (z^3 + \tfrac{1}{z}) + \tfrac{1}{8}(z^5 + \tfrac{1}{z^3})).
\end{aligned}$$

Notice that the subdivision scheme from $p(\omega)$ is also interpolatory. It is a 6-point C^2 interpolatory scheme.

Finally, let us consider the general case. With $z = e^{-i\omega}$, let $E(\omega)$ be the matrix of the form

$$E(\omega) = \begin{bmatrix} 1 & 0 & L_3(z) \\ 0 & 1 & L_4(z) \\ 0 & 0 & 1 \end{bmatrix}, \tag{87}$$

where $L_3(z)$ and $L_4(z)$ are Laurent polynomials satisfying $L_3(\frac{1}{z}) = zL_3(z), L_4(\frac{1}{z}) = zL_4(z)$, namely they are Laurent polynomials of the form (59). Clearly, $\widetilde{E}(\omega) = (E(\omega)^{-1})^*$ is

$$\widetilde{E}(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -L_3(\frac{1}{z}) & -L_4(\frac{1}{z}) & 1 \end{bmatrix}.$$

Then the type II frame filter banks corresponding to the algorithm with K $(K \geq 1)$ steps can be given as

$$[p(\omega),\ q^{(1)}(\omega),\ q^{(2)}(\omega)]^T = D_{K-1}(2\omega)D_{K-2}(2\omega)\cdots D_1(2\omega)D_0(2\omega)\begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix}, \tag{88}$$

$$[\widetilde{p}(\omega),\ \widetilde{q}^{(1)}(\omega),\ \widetilde{q}^{(2)}(\omega)]^T = \frac{1}{2}\widetilde{D}_{K-1}(2\omega)\widetilde{D}_{K-2}(2\omega)\cdots \widetilde{D}_1(2\omega)\widetilde{D}_0(2\omega)\begin{bmatrix} 1 \\ e^{-i\omega} \end{bmatrix}, \tag{89}$$

where D_0 and $\widetilde{D}_0$ are defined by (71) and (72), each $D_k(\omega)$, $1 \leq k \leq K-1$, is a matrix $E(\omega)$ of the form (87) or $\widetilde{E}(\omega)$, and $\widetilde{D}_k(\omega) = (D_k(\omega)^{-1})^*$. The next proposition shows that the framelets obtained by these algorithms have the uniform symmetry.

Proposition 2. *Let $\{p, q^{(1)}, q^{(2)}\}$ and $\{\widetilde{p}, \widetilde{q}^{(1)}, \widetilde{q}^{(2)}\}$ be the biorthogonal frame filter banks defined by (88) and (89). Then*

$$\begin{aligned} &p(-\omega) = p(\omega),\ q^{(1)}(-\omega) = q^{(1)}(\omega),\ q^{(2)}(-\omega) = e^{i2\omega}q^{(2)}(\omega), \\ &\widetilde{p}(-\omega) = \widetilde{p}(\omega),\ \widetilde{q}^{(1)}(-\omega) = \widetilde{q}^{(1)}(\omega),\ \widetilde{q}^{(2)}(-\omega) = e^{i2\omega}\widetilde{q}^{(2)}(\omega). \end{aligned}$$

Furthermore, the associated scaling functions $\phi, \widetilde{\phi}$, and framelets $\psi^{(\ell)}, \widetilde{\psi}^{(\ell)}, \ell = 1, 2$ satisfy

$$\begin{aligned} &\phi(x) = \phi(-x),\ \psi^{(1)}(x) = \psi^{(1)}(-x),\ \psi^{(2)}(x) = \psi^{(2)}(1-x), \\ &\widetilde{\phi}(x) = \widetilde{\phi}(-x),\ \widetilde{\psi}^{(1)}(x) = \widetilde{\psi}^{(1)}(-x),\ \widetilde{\psi}^{(2)}(x) = \widetilde{\psi}^{(2)}(1-x). \end{aligned}$$

The proof of Proposition 2 is essentially the same as that of Proposition 1. In this case one uses the fact that for $D(\omega) = D_k(\omega)$ or $D(\omega) = \widetilde{D}_k(\omega), k \geq 1$, $D(\omega)$ satisfies

$$D(-\omega) = \text{diag}(1, 1, e^{i\omega})D(\omega)\text{diag}(1, 1, e^{-i\omega}),$$

and the fact for $D(\omega) = D_0(\omega)$ or $D(\omega) = \widetilde{D}_0(\omega)$, $D(\omega)$ satisfies

$$D(-\omega) = \text{diag}(1, 1, e^{i\omega})D(\omega)\text{diag}(1, e^{-i\omega}).$$

References

[1] J.J. Benedetto and S.D. Li, The theory of multiresolution analysis frames and applications to filter banks, *Appl. Comput. Harmon. Anal.*, 5 (1998), 389–427.

[2] M. Bertram, Biorthogonal Loop-subdivision wavelets, *Computing*, 72 (2004), 29–39.

[3] M. Bownik, Tight frames of multidimensional wavelets, *J. Fourier Anal. Appl.*, 3 (1997), 525–542.

[4] J.-F. Cai, R.H. Chan, L.X. Shen, and Z.W. Shen, Restoration of chopped and nodded images by framelets, *SIAM J. Sci. Comput.*, 24 (2008), 1205–1227.

[5] R.H. Chan, T.F. Chan, L.X. Shen, and Z.W. Shen, Wavelet algorithms for high-resolution image reconstruction, *SIAM J. Sci. Comput.*, 24 (2003), 1408–1432.

[6] R.H. Chan, S.D. Riemenschneider, L.X. Shen, and Z.W. Shen, Tight frame: An efficient way for high-resolution image reconstruction, *Appl. Comput. Harmon. Anal.*, 17 (2004), 91–115.

[7] M. Charina and J. Stöckler, Tight wavelet frames for subdivision, *J. Comput. Appl. Math.*, 221 (2008), 293–301.

[8] M. Charina and J. Stöckler, Tight wavelet frames for irregular multiresolution analysis, *Appl. Comput. Harmon. Anal.*, 25 (2008), 98–113.

[9] O. Christensen, *An Introduction to Frames and Riesz Bases*, Birkhäuser, Boston, 2002.

[10] C.K. Chui and W.J. He, Compactly supported tight frames associated with refinable functions, *Appl. Comput. Harmon. Anal.*, 8 (2000), 293–319.

[11] C.K. Chui and W.J. He, Construction of multivariate tight frames via kronecker products, *Appl. Comput. Harmon. Anal.*, 11 (2001), 305-312.

[12] C.K. Chui, W.J. He, and J. Stöckler, Compactly supported tight and sibling frames with maximum vanishing moments, *Appl. Comput. Harmon. Anal.*, 13 (2002), 224-262.

[13] C.K. Chui, W.J. He, and J. Stöckler, Nonstationary tight wavelet frames. I. Bounded intervals, *Appl. Comput. Harmon. Anal.*, 17 (2004), 141-197.

[14] C.K. Chui, W.J. He, and J. Stöckler, Nonstationary tight wavelet frames. II. Unbounded intervals, *Appl. Comput. Harmon. Anal.*, 18 (2005), 25-66.

[15] C.K. Chui, W.J. He, J. Stöckler, and Q.Y. Sun, Compactly supported tight affine frames with integer dilations and maximum vanishing moments, *Adv. Comput. Math.*, 18 (2003), 159–187.

[16] C.K. Chui and X.L. Shi, Inequalities of Littlewood-Paley type for frames and wavelets, *SIAM J. Math. Anal.*, 24 (1993), 263–277.

[17] A. Cohen, I. Daubechies, and J.-C. Feauveau, Biorthogonal bases of compactly supported wavelets, *Comm. Pure Appl. Math.*, 45 (1992), 485–560.

[18] W. Dahmen, Decomposition of refinable spaces and applications to operator equations, *Numer. Algor.*, 5 (1993), 229–245.

[19] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics, Vol.61, SIAM, Philadelphia, PA, 1992.

[20] I. Daubechies and B. Han, The canonical dual frame of a wavelet frame, *Appl. Comput. Harmon. Anal.*, 12 (2002), 269–285.

[21] I. Daubechies, B. Han, A. Ron, and Z.W. Shen, Framelets: MRA-based construction of wavelet frames, *Appl. Comput. Harmon. Anal.*, 14 (2003), 1–46.

[22] B. Dong and Z.W. Shen, Pseudo-splines, wavelets and framelets, *Appl. Comput. Harmon. Anal.*, 22 (2007), 78–104.

[23] N. Dyn, J.A. Gregory and D. Levin, A 4-point interpolatory subdivision scheme for curve designs, *Comput. Aided Geom. Design*, 4 (1987), 257–268.

[24] M. Ehler, On multivariate compactly supported bi-frames, *J. Fourier Anal. Appl.*, 13 (2007), 511–532.

[25] T. Eirola, Sobolev characterization of solutions of dilation equations, *SIAM J. Math. Anal.*, 23 (1992), 1015–1030.

[26] K. Gröchenig, *Foundations of Time-Frequency Analysis*, Birkhäuser, Boston, 2001.

[27] K. Gröchenig and A. Ron, Tight compactly supported wavelet frames of arbitrary high smoothness, *Proc. Amer. Math. Soc.*, 126 (1998), 1101-1107.

[28] R. Gribonval and M. Nielsen, On approximation with spline generated framelets, *Constr. Approx.*, 20 (2004), 207–232.

[29] B. Han, On dual wavelet tight frames, *Appl. Comput. Harmon. Anal.*, 4 (1997), 380–413.

[30] B. Han, Compactly supported tight wavelet frames and orthonormal wavelets of exponential decay with a general dilation matrix, *J. Comput. Appl. Math.*, 155 (2003), 43–67.

[31] B. Han, Dual multiwavelet frames with high balancing order and compact fast frame transform, *Appl. Comput. Harmon. Anal.*, 26 (2009), 14–42.

[32] B. Han and Q. Mo, Multiwavelet frames from refinable function vectors, *Adv. Comput. Math.*, 18 (2003), 211–245.

[33] B. Han and Q. Mo, Symmetric MRA tight wavelet frames with three generators and high vanishing moments, *Appl. Comput. Harmon. Anal.*, 18 (2005), 67–93.

[34] B. Han and Z.W. Shen, Dual wavelet frames and Riesz bases in Sobolev spaces, *Constr. Approx.*, 29 (2009), 369–406.

[35] C. Heil and D. Walnut, Continuous and discrete wavelet transforms, *SIAM Rev.*, 31 (1989), 628–666.

[36] R.Q. Jia, Convergence of vector subdivision schemes and construction of biorthogonal multiple wavelets, In *Advances in Wavelets*, K.-S. Lau (ed.), Springer-Verlag, Singapore, 1999, pp. 199–227.

[37] R.Q. Jia and Q.T. Jiang, Spectral analysis of transition operators and its applications to smoothness analysis of wavelets, *SIAM J. Matrix Anal. Appl.*, 24 (2003), 1071–1109.

[38] R.Q. Jia and S.R. Zhang, Spectral properties of the transition operator associated to a multivariate refinement equation, *Linear Algebra Appl.*, 292 (1999), 155–178.

[39] Q.T. Jiang, Parameterizations of masks for tight affine frames with two symmetric/antisymmetric generators, *Adv. Comput. Math.*, 18 (2003), 247–268.

[40] Q.T. Jiang, Hexagonal tight frame filter banks with idealized high-pass filters, *Adv. Comput. Math.*, 31 (2009), 215–236.

[41] Q.T. Jiang, Biorthogonal wavelets with 6-fold axial symmetry for hexagonal data and triangle surface multiresolution processing, *International Journal of Wavelet, Multiresolution and Information Processing*, in press.

[42] Q.T. Jiang, Biorthogonal wavelets with 4-fold axial symmetry for quadrilateral surface multiresolution processing, *Adv. Comput. Math.*, in press.

[43] M.-J. Lai and J. Stöckler, Construction of multivariate compactly supported tight wavelet frames, *Appl. Comput. Harmon. Anal.*, 21 (2006), 324-348.

[44] J.M. Lounsbery, *Multiresolution Analysis for Surfaces of Arbitrary Topological Type*, Ph.D. Dissertation, University of Washington, Department of Mathematics, 1994.

[45] J.M. Lounsbery, T.D. Derose, and J. Warren, Multiresolution analysis for surfaces of arbitrary topological type, *ACM Trans. Graphics*, 16 (1997), 34–73.

[46] A. Petukhov, Explicit construction of framelets, *Appl. Comput. Harmon. Anal.*, 11 (2001), 313–327.

[47] A. Ron and Z.W. Shen, Affine systems in $L_2(\mathbb{R}^d)$: The analysis of the analysis operators, *J. Funct. Anal.*, 148 (1997), 408–447.

[48] A. Ron and Z.W. Shen, Affine systems in $L_2(\mathbb{R}^d)$ II: Dual systems, *J. Fourier Anal. Appl.*, 3 (1997), 617–637.

[49] A. Ron and Z.W. Shen, Compactly supported tight affine spline frames in $L_2(\mathbb{R}^d)$, *Math. Comput.*, 67 (1998), 191–207.

[50] A. Ron and Z.W. Shen, Construction of compactly supported affine spline frames in $L_2(\mathbb{R}^d)$, in *Advances in Wavelets*, K.-S. Lau (ed.), Springer-Verlag, Singapore, 1998, pp. 27–49.

[51] I.W. Selesnick, Smooth wavelet tight frames with zero moments, *Appl. Comput. Harmon. Anal.*, 10 (2001), 163–181.

[52] I.W. Selesnick and A.F. Abdelnour, Symmetric wavelet tight frames with two generators, *Appl. Comput. Harm. Anal.*, 17 (2004), 211–225.

[53] L.X. Shen, M. Papadakis, I.A. Kakadiaris, I.Konstantinidis, I. Kouri, and D. Hoffman, Image denoising using a tight frame, *IEEE Trans. Image Proc.*, 15 (2006), 1254–1263.

[54] E. Stollnitz, T. DeRose, and H. Salesin, *Wavelets for Computer Graphics*, Morgan Kaufmann Publishers, San Francisco, 1996.

[55] W. Sweldens, The lifting scheme: A custom-design construction of biorthogonal wavelets, *Appl. Comput. Harmon. Anal.*, 3 (1996), 186–200.

[56] H.W. Wang, K.H. Qin, and K. Tang, Efficient wavelet construction with Catmull-Clark subdivision, *The Visual Computer*, 22 (2006), 874–884.

[57] H.W. Wang, K.H. Qin, and H.Q. Sun, $\sqrt{3}$-subdivision-based biorthogonal wavelets, *IEEE Trans. Visualization and Computer Graphics*, 13 (2007), 914–925.

[58] L. Villemoes, Energy moments in time and frequency for two-scale difference equation solutions and wavelets, *SIAM J. Math. Anal.*, 23 (1992), 1519–1543.