Article

Construction of Multiwavelets on an Interval

Ahmet Altürk [1] **and Fritz Keinert** [2,]*****

[1] Department of Mathematics, Amasya University, Amasya, Turkey;
   E-Mail: ahmet.alturk@amasya.edu.tr

[2] Department of Mathematics, Iowa State University, Ames, IA 50011, USA

* Author to whom correspondence should be addressed; E-Mail: keinert@iastate.edu;
   Tel.: +39-366-501-8384; Fax: +39-051-2094490.

Abstract: Boundary functions for wavelets on a finite interval are often constructed as linear combinations of boundary-crossing scaling functions. An alternative approach is based on linear algebra techniques for truncating the infinite matrix of the Discrete Wavelet Transform to a finite one. In this article we show how an algorithm of Madych for scalar wavelets can be generalized to multiwavelets, given an extra assumption. We then develop a new algorithm that does not require this additional condition. Finally, we apply results from a previous paper to resolve the non-uniqueness of the algorithm by imposing regularity conditions (including approximation orders) on the boundary functions.

Keywords: wavelets on an interval; multiwavelets; discrete wavelet transform; boundary handling

1. Introduction and Overview

The Discrete Wavelet Transform (DWT) is designed to act on infinitely long signals. For finite signals, the algorithm breaks down near the boundaries. This can be dealt with by constructing special boundary functions [1–3], or by extending the data by zero padding, extrapolation, symmetry, or other methods [4–7].

Two approaches to constructing boundary functions were compared in a previous paper of the authors [8]. The first approach is based on forming linear combinations of standard scaling functions that cross the boundary. The second approach is based on boundary recursion relations.

A third approach is based on linear algebra. The infinite banded Toeplitz matrix representing the DWT is replaced by a finite matrix by suitable end-point modifications. A particular such method for scalar wavelets is presented in Madych [4].

In this paper, we first show that the Madych approach can be generalized to multiwavelets under an additional assumption, which may or may not be satisfied for a given multiwavelet. We then present a modified method that does not require this extra assumption.

Linear algebra completions are not unique; they all include multiplication by an arbitrary orthogonal matrix. A random choice of matrix does not in general produce coefficients that correspond to any actual boundary function. Random choice also does not provide any approximation order at the boundary.

We show how to impose approximation order constraints in the algorithm, and in the process remove much or all of the non-uniqueness.

2. Review of Wavelet Theory

In this section we provide a brief background on the theory of wavelets. We primarily focus on the basic definitions and results that will be used throughout this article. For a more detailed treatment, we refer the reader to the many excellent articles and books published on this subject [6,9–12].

We will state everything in terms of multiwavelets, which includes scalar wavelets as a special case, and restrict ourselves to the orthogonal case.

2.1. Multiresolution Approximation

A *multiresolution approximation* (MRA) of $L^2(\mathbb{R})$ is a chain of closed subspaces $\{V_n\}$, $n \in \mathbb{Z}$,

$$\cdots \subset V_{-1} \subset V_0 \subset V_1 \subset \cdots \subset L^2(\mathbb{R})$$

satisfying

  (i) $V_n \subset V_{n+1}$ for all $n \in \mathbb{Z}$;
 (ii) $f(x) \in V_n \Longleftrightarrow f(2x) \in V_{n+1}$ for all $n \in \mathbb{Z}$;
(iii) $f(x) \in V_n \Longrightarrow f(x - 2^{-n}k) \in V_n$ for all $n, k \in \mathbb{Z}$;
(iv) $\bigcap_{n \in \mathbb{Z}} V_n = \{0\}$;
 (v) $\overline{\bigcup_{n \in Z} V_n} = L^2(\mathbb{R})$;
(vi) there exists a function vector

$$\phi(x) = \begin{pmatrix} \phi_1(x) \\ \vdots \\ \phi_r(x) \end{pmatrix}, \qquad \phi_i \in L^2(\mathbb{R})$$

such that $\{\phi_j(x - k) : 1 \leq j \leq r, k \in \mathbb{Z}\}$ is an orthonormal basis for V_0 [11].

The function ϕ is called the *multiscaling function* of the given MRA. r is called the *multiplicity*.

Condition (ii) gives the main property of an MRA. Each V_n consists of the functions in V_0 compressed by a factor of 2^n. Thus, an orthonormal basis of V_n is given by

$$\{\phi_{j,nk} := 2^{n/2}\phi_j(2^n x - k) : 1 \leq j \leq r, k \in \mathbb{Z}\}$$

Since $V_0 \subset V_1$, ϕ can be written in terms of the basis of V_1 as

$$\phi(x) = \sum_k H_k \phi_{1k}(x) = \sqrt{2} \sum_k H_k \phi(2x - k)$$

for some $r \times r$ coefficient matrices H_k. This is called a *two-scale refinement equation*, and ϕ is called *refinable*. We consider in this paper only compactly supported functions, for which the refinement equation is a finite sum.

The orthogonal projection P_n of a function $s \in L^2$ into V_n is given by

$$P_n s = \sum_{k \in \mathbb{Z}} \langle s, \phi_{nk} \rangle \phi_{nk}$$

This is interpreted as an approximation to s at scale 2^{-n}.

Here the inner product is defined as

$$\langle f, g \rangle = \int f(x)\, g(x)^*\, dx$$

where $*$ denotes the complex conjugate transpose.

The main application of an MRA comes from considering the difference between approximations to s at successive scales 2^{-n} and 2^{-n-1}.

Let $Q_n = P_{n+1} - P_n$. Q_n is also an orthogonal projection onto a closed subspace W_n, which is the orthogonal complement of V_n in V_{n+1}:

$$V_{n+1} = V_n \oplus W_n$$

$Q_n s$ is interpreted as the fine detail in s at resolution 2^{-n}.

An orthonormal basis of W_0 is generated from the integer translates of a single function vector $\psi \in L^2(\mathbb{R})$, called a *multiwavelet function*. Since $W_0 \subset V_1$, the multiwavelet function ψ can be represented as

$$\psi(x) = \sqrt{2} \sum_n G_n \phi(2x - n) \tag{1}$$

for some coefficient matrices G_n.

We have

$$L^2(\mathbb{R}) = \bigoplus_{n \in \mathbb{Z}} W_n$$

and $\{\psi_{j,nk} : 1 \leq j \leq r, n, k \in \mathbb{Z}\}$ produces an orthonormal basis for $L^2(\mathbb{R})$.

2.2. Discrete Wavelet Transform

The *Discrete Wavelet Transform* (DWT) takes a function $s \in V_n$ for some n and decomposes it into a coarser approximation at level $m < n$, plus the fine detail at the intermediate levels.

$$s = P_n s = P_m s + \sum_{k=m}^{n-1} Q_n s$$

It suffices to describe the step from level n to level $n - 1$.

Since the signal $s \in V_n = V_{n-1} \oplus W_{n-1}$, we can represent it by its coefficients $\{\mathbf{s}^*_{nk}\} = \{\langle s, \boldsymbol{\phi}_{nk} \rangle\}$, $\{\mathbf{d}^*_{nk}\} = \{\langle s, \boldsymbol{\psi}_{nk} \rangle\}$ as

$$s = \sum_k \mathbf{s}^*_{nk} \boldsymbol{\phi}_{nk} = \sum_j \mathbf{s}^*_{n-1,j} \boldsymbol{\phi}_{n-1,j} + \sum_j \mathbf{d}^*_{n-1,j} \boldsymbol{\psi}_{n-1,j}$$

We find that

$$\mathbf{s}_{n-1,j} = \sum_k H_{k-2j} \mathbf{s}_{nk}, \qquad \mathbf{d}_{n-1,j} = \sum_k G_{k-2j} \mathbf{s}_{nk}$$

If we interleave the coefficients at level $n-1$

$$(\mathbf{sd})_{n-1} = \begin{pmatrix} \vdots \\ \mathbf{s}_{n-1,0} \\ \mathbf{d}_{n-1,0} \\ \mathbf{s}_{n-1,1} \\ \mathbf{d}_{n-1,1} \\ \vdots \end{pmatrix}$$

the DWT can be written as $(\mathbf{sd})_{n-1} = \Delta \mathbf{s}_n$, where

$$\Delta = \begin{pmatrix} \cdots & \cdots & \cdots & & & \\ \cdots & T_0 & T_1 & T_2 & \cdots & \\ & \cdots & T_0 & T_1 & T_2 & \cdots \\ & & \cdots & T_0 & T_1 & T_2 & \cdots \\ & & & \cdots & \cdots & \cdots \end{pmatrix}, \qquad T_k = \begin{pmatrix} H_{2k} & H_{2k+1} \\ G_{2k} & G_{2k+1} \end{pmatrix} \tag{2}$$

The matrix Δ is orthogonal. Signal reconstruction corresponds to $\mathbf{s}_n = \Delta^*(\mathbf{sd})_{n-1}$.

2.3. Approximation Order

A multiscaling function ϕ has *approximation order* p if all polynomials of degree less than p can be expressed locally as linear combinations of integer shifts of ϕ. That is, there exist row vectors $\mathbf{c}^*_{jk}$, $j = 0, \ldots, p-1$, $k \in \mathbb{Z}$, so that

$$x^j = \sum_k \mathbf{c}^*_{jk} \phi(x-k) \tag{3}$$

For orthogonal wavelets

$$\mathbf{c}^*_{jk} = \sum_{\ell=0}^{j} \binom{j}{\ell} k^{j-\ell} \boldsymbol{\mu}^*_\ell \tag{4}$$

where $\boldsymbol{\mu}_j$ is the jth continuous moment of ϕ.

A high approximation order is desirable in applications. A minimum approximation order of 1 is a required condition in many theorems.

3. Wavelets on an Interval

Standard wavelet theory only considers functions on the entire real line. In practice we often deal with functions on a finite interval I. One way to deal with this problem is to introduce special boundary basis functions. These functions need to be refinable in order to support a DWT algorithm.

The main approaches construct boundary functions from

- recursion relations; or
- linear combinations of shifts of the underlying scaling functions; or
- linear algebra techniques.

Details on these approaches, and the connections between them, will be given later in this section.

The linear combination approach has been the most commonly used technique (see e.g., [1,3,13,14]). The recursion relation approach, and its relationship to linear combinations, was studied in more detail in [8]. The linear algebra approach is used in [4,6].

3.1. Basic Assumptions and Notation

We do not aim for complete generality but make the following simplifying assumptions, which cover most cases of practical interest.

- The underlying multiwavelet is orthogonal, continuous, with multiplicity r and approximation order $p \geq 1$, and has recursion coefficients $H_0, \ldots, H_N$, $G_0, \ldots, G_N$. This means the support of ϕ and ψ is contained in the interval $[0, N]$.
- The interval I is $[0, M]$ with M large enough so that the left and right endpoint functions do not interfere with each other.
- The boundary functions have support on $[0, N-1]$ and $[M-N+1, M]$, respectively (that is, smaller support than the interior functions).

The *interior multiscaling functions* are those integer shifts of ϕ whose support fits completely inside $[0, M]$. These are $\phi_0, \ldots, \phi_{M-N}$, where $\phi_k(x) = \phi(x-k)$. All interior functions have value $\mathbf{0}$ at the endpoints, by continuity.

The interior functions satisfy the usual recursion relations

$$\phi(x) = \sqrt{2} \sum_k H_k \phi(2x - k)$$
$$\psi(x) = \sqrt{2} \sum_k G_k \phi(2x - k) \tag{5}$$

The *boundary-crossing multiscaling functions* are those shifts of ϕ whose support contains 0 or M in its interior. At the left endpoint, these are ϕ_{-N+1} through ϕ_{-1}.

The support of ϕ could be strictly smaller than $[0, N]$. In this case, some of the functions that appear to be boundary-crossing are actually interior, but this causes no problems.

We assume that we have L left endpoint scaling functions, which we group together into a single vector ϕ^L. We stress that we mean L scalar functions, not function vectors, and that L is not necessarily a multiple of r.

Likewise, we assume R right endpoint functions with support contained in $[M - N + 1, M]$, grouped into a vector ϕ^R, with recursion relations similar to (6). We will show later that L and R are uniquely determined by ϕ.

Most of the rest of the paper will only address the calculations at the left end in detail. Calculations at the right end are identical, with appropriate minor changes.

3.2. Recursion Relations

The left endpoint functions satisfy recursion relations

$$
\begin{aligned}
\phi^L(x) &= \sqrt{2}A\phi^L(2x) + \sqrt{2}\sum_{k=0}^{N-2} B_k\phi(2x - k) \\
\psi^L(x) &= \sqrt{2}E\phi^L(2x) + \sqrt{2}\sum_{k=0}^{N-2} F_k\phi(2x - k)
\end{aligned}
\tag{6}
$$

Here

$$
A = \langle \phi_{n-1}^L, \phi_n^L \rangle, \quad B_k = \langle \phi_{n-1}^L, \phi_{n,k} \rangle, \quad E = \langle \psi_{n-1}^L, \phi_n^L \rangle, \quad F_k = \langle \psi_{n-1}^L, \phi_{n,k} \rangle
$$

where $\phi_n^L(x) = 2^{n/2}\phi^L(2^n x)$. A and E are of size $L \times L$, B_k and F_k are of size $L \times r$.

The recursion relation approach constructs boundary functions by finding suitable recursion coefficients A, B.

We are interested in *regular solutions* of (6), that is, ϕ^L that are continuous, have approximation order at least 1, and $\phi^L(0) \neq \mathbf{0}$.

It is shown in [8] that a sufficient condition for regularity is that A has a simple largest eigenvalue of $1/\sqrt{2}$, and that ϕ^L has approximation order at least 1. Conditions for verifying boundary approximation order p are given in Section 6.

3.3. Linear Combinations

If the boundary functions are linear combinations of boundary-crossing functions, then

$$
\phi^L(x) = \sum_{k=-N+1}^{-1} C_k\phi(x - k) \qquad \text{for } x > 0
\tag{7}
$$

Each C_k is of size $L \times r$.

The linear combination approach to constructing boundary function tries to construct a suitable C. A random choice of C_k will not produce refinable functions. It is shown in [8] that if the boundary functions are both refinable and linear combinations, the coefficients must be related by

$$
\begin{aligned}
CV &= AC \\
CW &= B
\end{aligned}
\tag{8}
$$

where

$$
B = \begin{pmatrix} B_0 & B_1 & \cdots & B_{N-2} \end{pmatrix}, \qquad C = \begin{pmatrix} C_{-N+1} & C_{-N+2} & \cdots & C_{-1} \end{pmatrix}
$$

and

$$\left(V\middle|W\right) = \left(\begin{array}{cccccc|cccccc} H_{N-1} & H_N & 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ H_{N-3} & H_{N-2} & H_{N-1} & H_N & 0 & 0 & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & H_3 & H_4 & \cdots & \cdots & H_N & 0 & 0 \\ 0 & \cdots & \cdots & 0 & H_0 & H_1 & H_2 & \cdots & \cdots & H_{N-2} & H_{N-1} & H_N \end{array}\right)$$

Both V and W are of size $(N-1)r \times (N-1)r$.

Relations (8) are a kind of eigenvalue problem. For any given ϕ there is only a small number of possibilities for boundary functions that are both refinable and linear combinations.

Note that the recursion relations are necessary. Without coefficients A, B there is no discrete wavelet transform. On the other hand, the existence of C as in (7) is optional. It was shown in [8] that there are regular refinable boundary functions that are not linear combinations.

3.4. Linear Algebra

We can assume that $N = 2K + 1$ is odd, by introducing an extra recursion coefficient $H_N = 0$ if necessary. The resulting structure for the decomposition matrix is

$$\Delta_M = \left(\begin{array}{c|ccccccc|c} L_0 & L_1 & \cdots & L_K & 0 & \cdots & \cdots & 0 & 0 \\ \hline 0 & T_0 & T_1 & \cdots & T_K & 0 & \cdots & 0 & 0 \\ \vdots & 0 & T_0 & T_1 & \ddots & T_K & 0 & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & T_0 & T_1 & \cdots & T_K & 0 \\ \hline 0 & 0 & \cdots & \cdots & 0 & R_0 & R_1 & \cdots & R_K \end{array}\right) \tag{9}$$

This corresponds to a segment of the infinite matrix Δ in (2) with some end point modifications.

Here the T_k are as in (2), and

$$L_0 = \begin{pmatrix} A \\ E \end{pmatrix}, \quad L_k = \begin{pmatrix} B_{2k-2} & B_{2k-1} \\ F_{2k-2} & F_{2k-1} \end{pmatrix}, \quad k = 1, \ldots, K \tag{10}$$

with A, B, E, F as in (6).

The linear algebra approach tries to construct suitable endpoint blocks L_k, R_k by linear algebra methods.

3.5. Uniqueness Results

We assume initially that there are only four recursion coefficients, and therefore only two block matrices

$$T_0 = \begin{pmatrix} H_0 & H_1 \\ G_0 & G_1 \end{pmatrix}, \quad T_1 = \begin{pmatrix} H_2 & H_3 \\ G_2 & G_3 \end{pmatrix}$$

Since the infinite matrix Δ in (2) is orthogonal, we know that

$$\begin{aligned} T_0 T_0^* + T_1 T_1^* &= I \\ T_0 T_1^* &= 0 \end{aligned} \tag{11}$$

or equivalently

$$T_0^* T_0 + T_1^* T_1 = I$$
$$T_0^* T_1 = 0 \tag{12}$$

These relations lead to some interesting properties that we use in the following. More detail is given in [8,15], but for convenience we give at least an outline of the proofs here.

Lemma 3.1 *If T_0, T_1 are square matrices of size $2r \times 2r$ that satisfy relations (11), then*

(a) $\rho_0 := rank(T_0)$ and $\rho_1 := rank(T_1)$ satisfy

$$\rho_0 + \rho_1 = 2r$$

(b) The ranges and nullspaces of T_0 and T_1 are mutually orthogonal and complementary, that is,

$$R(T_0) \oplus R(T_1) = \mathbb{R}^{2r}$$
$$N(T_0) \oplus N(T_1) = \mathbb{R}^{2r}$$

(c) There exist orthogonal matrices U, V with

$$T_0 = U \begin{pmatrix} I_{\rho_0} & 0 \\ 0 & 0 \end{pmatrix} V^*, \qquad T_1 = U \begin{pmatrix} 0 & 0 \\ 0 & I_{\rho_1} \end{pmatrix} V^* \tag{13}$$

where I_ρ denotes an identity matrix of size $\rho \times \rho$.

Proof.

(a) The first equation in (11) implies $\rho_0 + \rho_1 \geq 2r$. The second equation implies $\rho_0 + \rho_1 \leq 2r$.

(b) The relation $T_0^* T_1 = 0$ implies that $R(T_1)$ is contained in $N(T_0^*) = R(T_0)^\perp$. The dimension count shows that they are identical.

(c) We start with separate singular value decompositions (SVD) of T_0 and T_1

$$T_0 = U_0 \begin{pmatrix} \Sigma_0 & 0 \\ 0 & 0 \end{pmatrix} V_0^*, \qquad T_1 = U_1 \begin{pmatrix} 0 & 0 \\ 0 & \Sigma_1 \end{pmatrix} V_1^*$$

where the usual ordering of singular values has been reversed for T_1.

Since ranges and nullspaces are orthogonal, we can construct matrices U and V by taking the first ρ_0 columns of U_0, V_0 and last ρ_1 columns of U_1, V_1, respectively, which provide a common SVD. The fact that $\Sigma_0 = I$, $\Sigma_1 = I$ follows from

$$(T_0 + T_1)(T_0^* + T_1^*) = \begin{pmatrix} \Sigma_0^2 & 0 \\ 0 & \Sigma_1^2 \end{pmatrix} = I$$

■ To investigate the possible sizes of the endpoint blocks in Δ_M, it suffices to consider

$$\Delta_3 = \begin{pmatrix} L_0 & L_1 & 0 & 0 \\ 0 & T_0 & T_1 & 0 \\ 0 & 0 & R_0 & R_1 \end{pmatrix} \tag{14}$$

of size $6r \times 6r$. Δ_M for larger M simply has more rows of T_0, T_1 in the middle.

Theorem 3.2 *If Δ_3 is orthogonal and has the structure given in* (14)*, then L_0, L_1, R_0, R_1 must have sizes $2\rho_1 \times \rho_1$, $2\rho_1 \times 2r$, $2\rho_0 \times 2r$, and $2\rho_0 \times \rho_0$, respectively.*

Theorem 3.3 *Assume that Δ_3, $\tilde{\Delta}_3$ are two orthogonal matrices of form* (14)*. Then there exist orthogonal matrices Q_L, Q_R so that*

$$\tilde{\Delta}_3 = \begin{pmatrix} Q_L & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & Q_R \end{pmatrix} \Delta_3 \tag{15}$$

The proofs are given in [8].

If there are more than two matrices T_j, we form block matrices. For example if we have $T_0, \ldots, T_3$, we use

$$\hat{T}_0 = \begin{pmatrix} T_0 & T_1 & T_2 \\ 0 & T_0 & T_1 \\ 0 & 0 & T_0 \end{pmatrix}, \qquad \hat{T}_1 = \begin{pmatrix} T_3 & 0 & 0 \\ T_2 & T_3 & 0 \\ T_1 & T_2 & T_3 \end{pmatrix} \tag{16}$$

4. Madych Approach for Scalar Wavelets

This is a particular implementation of the matrix completion approach for scalar orthogonal wavelets. Madych [4] started from a periodized version of the infinite matrix (2), and modified it into the desired form by orthogonal matrices. We will show in this section that this also works for orthogonal multiwavelets, given an additional condition.

To explain the Madych algorithm, and simultaneously extend it to multiwavelets, we again assume initially that we only have T_0 and T_1, and begin with

$$\tilde{\Delta}_3 = \begin{pmatrix} T_0 & T_1 & 0 \\ 0 & T_0 & T_1 \\ T_1 & 0 & T_0 \end{pmatrix} \tag{17}$$

Our objective is to find orthogonal matrices U, V so that

$$\Delta_3 = U\tilde{\Delta}_3 V = \begin{pmatrix} L_0 & L_1 & 0 & 0 \\ 0 & T_0 & T_1 & 0 \\ 0 & 0 & R_0 & R_1 \end{pmatrix}$$

has the desired structure and is orthogonal.

We let

$$S_0 = \begin{pmatrix} H_0 & H_1 \end{pmatrix}, \qquad S_1 = \begin{pmatrix} H_2 & H_3 \end{pmatrix}, \qquad W_0 = \begin{pmatrix} G_0 & G_1 \end{pmatrix}, \qquad W_1 = \begin{pmatrix} G_2 & G_3 \end{pmatrix}$$

From (11) it follows that

$$T_0 S_1^* = T_0 W_1^* = T_1 S_0^* = T_1 W_0^* = 0$$

Definition 4.1 *A multiscaling function based on four recursion coefficients satisfies* Condition M *if*

$$\begin{pmatrix} S_0 \\ S_1 \end{pmatrix} = \begin{pmatrix} H_0 & H_1 \\ H_2 & H_3 \end{pmatrix}$$

has full rank.

This condition is automatic in the scalar case, because S_0 and S_1 are non-zero orthogonal row vectors satisfying $S_0 S_1^* = 0$.

For multiwavelets this condition may not hold. For example, the Chui–Lian CL(2) multiwavelet [16] has only three recursion coefficients

$$H_0 = \frac{1}{4\sqrt{2}} \begin{pmatrix} 2 & 2 \\ -\sqrt{7} & -\sqrt{7} \end{pmatrix}, \quad H_1 = \frac{1}{4\sqrt{2}} \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}, \quad H_2 = \frac{1}{4\sqrt{2}} \begin{pmatrix} 2 & -2 \\ \sqrt{7} & -\sqrt{7} \end{pmatrix}$$

and S_1 only has rank 1.

The row spans of S_0, S_1 are mutually orthogonal, so we can orthonormalize them separately. We can find $r \times r$ nonsingular matrices R_0, R_1 so that

$$\begin{pmatrix} R_0 S_0 \\ R_1 S_1 \end{pmatrix}$$

is an orthogonal matrix.

We now let

$$V = \begin{pmatrix} (R_0 S_0)^* & 0 & (R_1 S_1)^* \\ 0 & I & 0 \end{pmatrix}$$

then

$$\tilde{\Delta}_3 V = \begin{pmatrix} T_0 (R_0 S_0)^* & T_1 & 0 & T_0 (R_1 S_1)^* \\ 0 & T_0 & T_1 & 0 \\ T_1 (R_0 S_0)^* & 0 & T_0 & T_1 (R_1 S_1)^* \end{pmatrix} = \begin{pmatrix} T_0 (R_0 S_0)^* & T_1 & 0 & 0 \\ 0 & T_0 & T_1 & 0 \\ 0 & 0 & T_0 & T_1 (R_1 S_1)^* \end{pmatrix}$$

This already has the desired form. Multiply $\tilde{\Delta}_3 V$ from the left by

$$U = \begin{pmatrix} U_L & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & U_R \end{pmatrix}$$

where U_L, U_R are arbitrary orthogonal matrices, to obtain

$$\Delta_3 = U \tilde{\Delta}_3 V = \begin{pmatrix} L_0 & L_1 & 0 & 0 \\ 0 & T_0 & T_1 & 0 \\ 0 & 0 & R_0 & R_1 \end{pmatrix}$$

with

$$\begin{aligned} L_0 &= U_L T_0 S_0^* R_0^* \\ L_1 &= U_L T_1 \\ R_0 &= U_R T_0 \\ R_1 &= U_R T_1 S_1^* R_1^* \end{aligned}$$

This completes the algorithm in the case of up to four recursion coefficients. In the general case we form block matrices again, as in (16).

5. A New Approach to Multiwavelet Endpoint Modification

For a more general algorithm, we again begin with $\tilde{\Delta}_3$ as in (17).

Let U and V be the orthogonal matrices from the joint SVD of T_0 and T_1 in (13). Consider

$$U_3^* = \begin{pmatrix} U^* & 0 & 0 \\ 0 & U^* & 0 \\ 0 & 0 & U^* \end{pmatrix}, \qquad V_3 = \begin{pmatrix} V & 0 & 0 \\ 0 & V & 0 \\ 0 & 0 & V \end{pmatrix}$$

We obtain

$$U_3^* \tilde{\Delta}_3 V_3 = \left(\begin{array}{cc|cc|cc} I_{\rho_0} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{\rho_1} & 0 & 0 \\ \hline 0 & 0 & I_{\rho_0} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{\rho_1} \\ \hline 0 & 0 & 0 & 0 & I_{\rho_0} & 0 \\ 0 & I_{\rho_1} & 0 & 0 & 0 & 0 \end{array} \right) \tag{18}$$

By inspection, a technique that produces the correct structure is to move the first ρ_0 columns to the end, and then interchange the first ρ_0 rows with the last ρ_1 rows. That amounts to multiplying from the right with

$$P_R = \begin{pmatrix} 0 & I_{\rho_0} \\ I_{6r-\rho_0} & 0 \end{pmatrix}$$

and from the left with

$$P_L = \begin{pmatrix} 0 & 0 & I_{\rho_0} \\ 0 & I_{4r} & 0 \\ I_{\rho_1} & 0 & 0 \end{pmatrix}$$

so that

$$P_L U_3^* \tilde{\Delta}_3 V_3 P_R = \left(\begin{array}{c|cc|cc|c} I_{\rho_1} & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & I_{\rho_1} & 0 & 0 & 0 \\ \hline 0 & I_{\rho_0} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{\rho_1} & 0 \\ \hline 0 & 0 & 0 & I_{\rho_0} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{\rho_0} \end{array} \right) \tag{19}$$

Now let Q_L, Q_R be arbitrary orthogonal matrices of size $2\rho_1 \times 2\rho_1, 2\rho_0 \times 2\rho_0$, respectively. We multiply (19) from the left with

$$U_L = \begin{pmatrix} Q_L & 0 & 0 \\ 0 & U & 0 \\ 0 & 0 & Q_R \end{pmatrix}$$

and from the right with

$$U_R = \begin{pmatrix} I_{\rho_1} & 0 & 0 & 0 \\ 0 & V^* & 0 & 0 \\ 0 & 0 & V^* & 0 \\ 0 & 0 & 0 & I_{\rho_0} \end{pmatrix}$$

then

$$\Delta_3 = U_L P_L U_3^* \tilde{\Delta}_3 V_3 P_R U_R = \begin{pmatrix} L_0 & L_1 & 0 & 0 \\ 0 & T_0 & T_1 & 0 \\ 0 & 0 & R_0 & R_1 \end{pmatrix}$$

where

$$\begin{aligned} L_0 &= Q_{L,0} \\ L_1 &= Q_{L,1} V_1^* \\ R_0 &= Q_{R,0} V_0^* \\ R_1 &= Q_{R,1} \end{aligned}$$

Here

$$Q_L = (Q_{L,0} \,|\, Q_{L,1}), \qquad Q_R = (Q_{R,0} \,|\, Q_{R,1}), \qquad V = (V_0 \,|\, V_1) \tag{20}$$

with $Q_{L,0}$, $Q_{L,1}$ of size $2\rho_1 \times \rho_1$, $Q_{R,0}$, $Q_{R,1}$ of size $2\rho_0 \times \rho_0$, and V_0, V_1 of size $2r \times \rho_0$, $2r \times \rho_1$, respectively.

There is one important observation that follows from Theorem 3.3. Suppose that $\tilde{Q}_L$, $\tilde{Q}_R$ are orthogonal matrices, and instead of (15) we define

$$\tilde{\Delta}_3 = \Delta_3 \begin{pmatrix} \tilde{Q}_L & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \tilde{Q}_R \end{pmatrix}$$

Then $\tilde{\Delta}_3$ has the form (14) and is orthogonal. By Theorem 3.3, there are other orthogonal matrices Q_L, Q_R so that the same $\tilde{\Delta}_3$ can also be written in the form (15). For this reason, it is not necessary to consider arbitrary orthogonal components in the matrix U_R.

As before, in the case of more than four recursion coefficients we apply the algorithm to block matrices.

If we reconsider Madych's approach in our notation, we find that it amounts to moving the second set of columns in (18) to the end, so that we end up with

$$\begin{pmatrix} I_{\rho_0} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I_{\rho_1} & 0 & 0 & 0 \\ 0 & I_{\rho_0} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{\rho_1} & 0 \\ 0 & 0 & 0 & I_{\rho_0} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{\rho_1} \end{pmatrix}$$

This only yields matrices of the correct size if $\rho_0 = \rho_1$, which is equivalent to Condition M.

6. Imposing Regularity Conditions

The algorithm in this section will only be described in detail for the left boundary. Calculations at the right boundary work the same way.

Given a completion Δ_M we can read off the recursion coefficients A, B from L_0, L_1. During numerical experiments with the algorithm described above we discovered that a random choice for the matrix Q_L did not usually correspond to actual boundary functions. This section will describe how to make a canonical choice for Q_L which produces recursion coefficients that correspond to regular boundary functions, and which imposes some approximation orders. The groundwork for this was laid in [8].

6.1. Approximation Order for Boundary Functions

In [8] it was shown that if the interior multiscaling function ϕ has approximation order $\geq p$, then approximation order p for the boundary scaling functions is equivalent to the existence of row vectors $\boldsymbol{\ell}_j^*$, $j = 0, \ldots, p-1$ so that

$$
\begin{aligned}
\boldsymbol{\ell}_j^* \left(\sqrt{2}A \right) &= 2^{-j} \boldsymbol{\ell}_j^* \\
\boldsymbol{\ell}_j^* \left(\sqrt{2}B_m \right) &= \boldsymbol{\gamma}_{jm}^*, \quad m = 0, \ldots, N-2
\end{aligned}
\tag{21}
$$

where the $\boldsymbol{\gamma}_{jm}^*$ are known row vectors:

$$
\boldsymbol{\gamma}_{jm}^* = 2^{-j} \mathbf{c}_{jm}^* - \sqrt{2} \sum_{k=0}^{\lfloor m/2 \rfloor} \mathbf{c}_{jk}^* H_{m-2k}
$$

Here $\lfloor x \rfloor$ = greatest integer $\leq x$, and $\mathbf{c}_{jk}^*$ are defined in (3).

If we let

$$
\boldsymbol{\gamma}_j^* = \left(\boldsymbol{\gamma}_{j0}^*, \ldots, \boldsymbol{\gamma}_{j,N-2}^* \right), \qquad
\Gamma = \begin{pmatrix} \boldsymbol{\gamma}_0^* \\ \vdots \\ \boldsymbol{\gamma}_{p-1}^* \end{pmatrix}, \qquad
\Lambda = \begin{pmatrix} \boldsymbol{\ell}_0^* \\ \vdots \\ \boldsymbol{\ell}_{p-1}^* \end{pmatrix}
$$

the approximation order conditions can be written as

$$
\begin{aligned}
\Lambda \left(\sqrt{2}A \right) &= \begin{pmatrix} 1 & 0 & \cdots & & 0 \\ 0 & 1/2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & 0 \\ 0 & \cdots & & 0 & 2^{-p+1} \end{pmatrix} \Lambda \\
\Lambda \left(\sqrt{2}B \right) &= \Gamma
\end{aligned}
\tag{22}
$$

After applying the algorithm from Section 5 with initial multiplier $Q_L = I$, we end up with

$$
\begin{pmatrix} L_0 & L_1 \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & V_1^* \end{pmatrix}
$$

with V_1 as in (20). After pre-multiplying by a general Q_L, we get

$$
Q_L \begin{pmatrix} L_0 & L_1 \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & V_1^* \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12}V_1^* \\ Q_{21} & Q_{22}V_1^* \end{pmatrix} = \begin{pmatrix} A & B \\ D & E \end{pmatrix}
$$

so $A = Q_{11}$, $B = Q_{12}V_1^*$.

The second condition in (22) is

$$\Lambda\left(\sqrt{2}B\right) = \Lambda\left(\sqrt{2}Q_{12}V_1^*\right) = \Gamma$$

We recall that $V_1^*V_1 = I$ and multiply by V_1 to get

$$\Lambda\sqrt{2}Q_{12} = \Lambda\sqrt{2}Q_{12}V_1^*V_1 = \Gamma V_1 =: G$$

where G is a known matrix with rows $\mathbf{g}_j^*$.

Conditions (22) reduce to

$$\Lambda\left(\sqrt{2}Q_{11}\right) = \begin{pmatrix} 1 & 0 & \cdots & & 0 \\ 0 & 1/2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & 0 \\ 0 & \cdots & & 0 & 2^{-p+1} \end{pmatrix}\Lambda$$

$$\Lambda\left(\sqrt{2}Q_{12}\right) = G$$

$$Q_{11}Q_{11}^* + Q_{12}Q_{12}^* = I$$

(23)

6.2. Simplifying the Problem

It is easy to see that if we replace the boundary function vector ϕ^L by $M\phi^L$ for an invertible matrix M, the new boundary functions still span the same space, and remain refinable. They also remain orthogonal if M is orthogonal.

The effect of M on the coefficients A, B, C and the matrix Λ is

$$A \to \tilde{A} = MAM^{-1}$$
$$B \to \tilde{B} = MB$$
$$C \to \tilde{C} = MC$$
$$\Lambda \to \tilde{\Lambda} = \Lambda M^{-1}$$

The key observation is that by using a suitable M, we can assume that Λ is lower triangular.

6.3. Deriving the Algorithm

We will now satisfy conditions (23) row by row.

NOTE: The vectors $\boldsymbol{\ell}_j^*$, $\boldsymbol{\gamma}_j^*$, $\mathbf{g}_j^*$ are naturally numbered $j = 0, \ldots, p - 1$. To keep the notation readable, in this section we will number the elements of vectors and matrices starting with index 0 instead of 1.

We start with row 0. The assumptions $\boldsymbol{\ell}_0^* = (l_{00}, 0, \ldots, 0)$ and $\boldsymbol{\ell}_0^*(\sqrt{2}Q_{11}) = \boldsymbol{\ell}_0^*$ lead to

$$Q_{11} = \begin{pmatrix} 1/\sqrt{2} & 0 & \cdots & 0 \\ * & \cdots & \cdots & * \\ \vdots & & & \vdots \\ * & \cdots & \cdots & * \end{pmatrix}$$

where $*$ denotes as yet undetermined entries.

The condition

$$\boldsymbol{\ell}_0^* \sqrt{2} Q_{12} = \mathbf{g}_0^*$$

leads to

$$Q_{12} = \frac{1}{\sqrt{2} l_{00}} \begin{pmatrix} \mathbf{g}_0^* \\ * \end{pmatrix}$$

The 00-entry of $Q_{11} Q_{11}^* + Q_{12} Q_{12}^*$ is

$$\frac{1}{2} + \frac{1}{2 l_{00}^2} \|\mathbf{g}_0^*\|^2 = 1$$

which leads to

$$l_{00} = \pm \|\mathbf{g}_0^*\|$$

It follows from the results in [8] that $l_{00} \neq 0$.

We have found the 0th row of Q_L, as well as $\boldsymbol{\ell}_0^*$.

For the induction step, assume that rows $0, \ldots, k-1$ of Λ, Q_L have already been determined. We partition the matrix Q_L as follows:

$$Q_L = \left(\begin{array}{cccc|cccc|cc} & Q_{11,k} & & & 0 & & \mathbf{0}^* & & \multicolumn{2}{c}{Q_{12,k}} \\ \hline q_{k,0} & \cdots & q_{k,k-1} & q_{k,k} & q_{k,k+1} & \cdots & q_{k,\rho_1-1} & q_{k,\rho_1} & \cdots & q_{k,2*\rho_1-1} \\ \hline & * & & * & & * & & & * & \end{array} \right)$$

$$= \left(\begin{array}{c|c|c|c} Q_{11,k} & 0 & \mathbf{0}^* & Q_{12,k} \\ \hline \boldsymbol{\alpha}_k^* & q_{k,k} & \boldsymbol{\zeta}_k^* & \boldsymbol{\beta}_k^* \\ \hline * & * & * & * \end{array} \right)$$

where $Q_{11,k}$ is lower triangular, and likewise

$$\boldsymbol{\ell}_k^* = \left(\begin{array}{ccc|c|ccc} l_{k0} & \cdots & l_{k,k-1} & l_{kk} & 0 & \cdots & 0 \end{array} \right)$$

$$= \left(\begin{array}{c|c|c} \boldsymbol{\lambda}_k^* & l_{kk} & \mathbf{0}^* \end{array} \right)$$

Note that

$$Q_{11,k} Q_{11,k}^* + Q_{12,k} Q_{12,k}^* = I$$

by induction. We now satisfy the conditions for the kth row in (23) one by one.

Assuming $l_{kk} \neq 0$, the condition $\boldsymbol{\ell}_k^* (\sqrt{2} Q_{11}) = 2^{-k} \boldsymbol{\ell}_k^*$ leads to

$$\boldsymbol{\zeta}_k^* = \mathbf{0}^*$$
$$\boldsymbol{\alpha}_k^* = \frac{1}{\sqrt{2} l_{kk}} \boldsymbol{\lambda}_k^* \left(2^{-k} I - \sqrt{2} Q_{11,k} \right) \tag{24}$$
$$q_{k,k} = 2^{-k-1/2}$$

The condition $\boldsymbol{\ell}_k^* (\sqrt{2} Q_{12}) = \mathbf{g}_k^*$ leads to

$$\boldsymbol{\beta}_k^* = \frac{1}{\sqrt{2} l_{kk}} \left(\mathbf{g}_k^* - \sqrt{2} \boldsymbol{\lambda}_k^* Q_{12,k} \right) \tag{25}$$

The new row k of Q must be orthogonal to the rows of $Q_{11,k}$

$$\mathbf{0}^* = \left(\; \boldsymbol{\alpha}_k^* \; \middle| \; q_{k,k} \; \middle| \; \mathbf{0}^* \; \right) \begin{pmatrix} Q_{11,k}^* \\ 0 \\ \mathbf{0}^* \end{pmatrix} + \boldsymbol{\beta}_k^* Q_{12,k}^*$$

$$= \boldsymbol{\alpha}_k^* Q_{11,k}^* + \boldsymbol{\beta}_k^* Q_{12,k}^*.$$

We substitute (24) and (25) and simplify to get

$$\boldsymbol{\lambda}_k^* = \mathbf{g}_k^* Q_{12,k}^* \left(\sqrt{2} I - 2^{-k} Q_{11,k}^* \right)^{-1} \tag{26}$$

Note that the matrix in parentheses is upper triangular with non-zero diagonal entries, and therefore non-singular.

Finally, we normalize the new row k

$$1 = \boldsymbol{\alpha}_k^* \boldsymbol{\alpha}_k + q_{kk}^2 + \boldsymbol{\beta}_k^* \boldsymbol{\beta}_k$$

which leads to

$$l_{kk}^2 = \frac{1}{2(1 - 2^{-2k-1})} \left[\boldsymbol{\lambda}_k^* \left(2^{-k} I - \sqrt{2} Q_{11,k} \right) \left(2^{-k} I - \sqrt{2} Q_{11,k}^* \right) \boldsymbol{\lambda}_k \right. \\ \left. + \left(\mathbf{g}_k^* - \sqrt{2} \boldsymbol{\lambda}_k^* Q_{12,k} \right) \left(\mathbf{g}_k - \sqrt{2} Q_{12,k}^* \boldsymbol{\lambda}_k \right) \right] \tag{27}$$

The actual calculations go in the following order: First we calculate $\boldsymbol{\lambda}_k^*$ and l_{kk} from (26) and (27). Then we can calculate $\boldsymbol{\alpha}_k^*$, $\boldsymbol{\beta}_k^*$ and q_{kk} from (24) and (25). Everything is unique, except for the choice of sign in l_{kk}.

The only way in which the algorithm could fail is if $l_{kk} = 0$. This is conceivable, but has not been observed in practice.

With this algorithm we can impose boundary approximation orders up to the number of boundary functions L, or the interior approximation order p, whichever is smaller. If $L < p$, the boundary functions will have a lower approximation order than the interior functions. If $L > p$, only the top p rows of A, B will be determined, and we have to make some arbitrary choices about the rest.

This will be illustrated by examples in the next section.

7. Examples

To keep the subscripting manageable, we assume that for left endpoint calculations we are working on the interval $[0, M]$. For right endpoint calculations we instead use $[-M, 0]$. The right-endpoint formulas corresponding to (6) and (7) are

$$\phi^R(x) = \sqrt{2} \sum_{k=-N+2}^{0} Y_k \phi(2x - k) + \sqrt{2} Z \phi^R(2x)$$

$$\phi^R(x) = \sum_{k=1}^{N-1} X_k \phi(x - k) \qquad \text{for } x < 0$$

The examples only show the calculations for ϕ^L, ϕ^R. The recursion coefficients for the wavelet functions ψ^L, ψ^R are found by completing the orthogonal matrices Q_L and Q_R.

7.1. Example 1: Daubechies D_2

As a very simple scalar example we consider the Daubechies scaling functions with two vanishing moments.

The recursion coefficients are

$$h_0 = \frac{1 + \sqrt{3}}{4\sqrt{2}}, \quad h_1 = \frac{3 + \sqrt{3}}{4\sqrt{2}}, \quad h_2 = \frac{3 - \sqrt{3}}{4\sqrt{2}}, \quad h_3 = \frac{1 - \sqrt{3}}{4\sqrt{2}}$$

The approximation order is 2, but there is only a single boundary function at each end. We can impose approximation order 1 at each end.

At the left end, we find
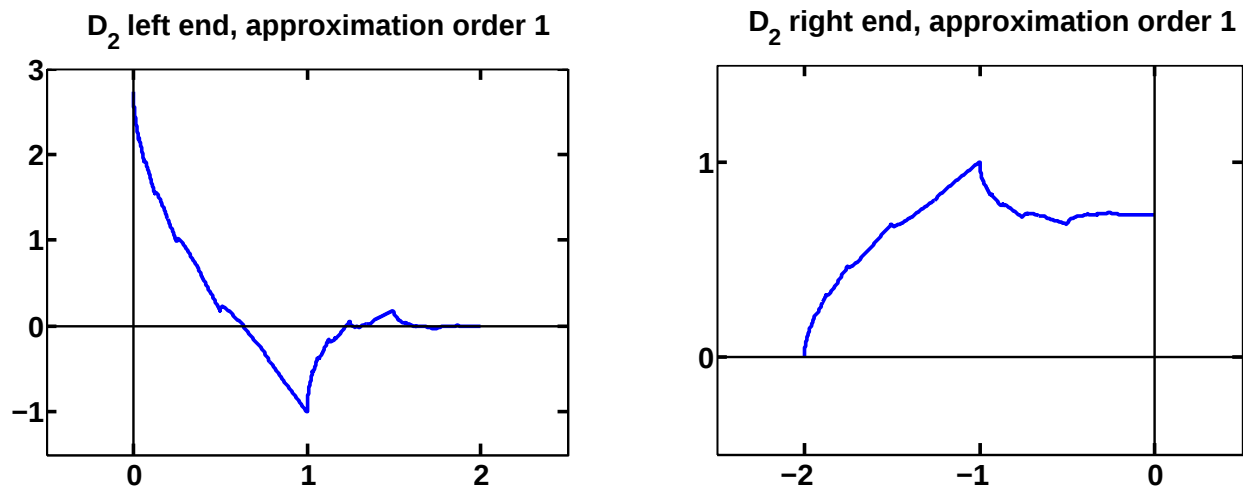
$$a = \frac{1}{2}\sqrt{2}, \qquad \mathbf{b}^* = \left(\frac{\sqrt{6}}{4}, \ -\frac{\sqrt{2}}{4} \right), \qquad \mathbf{c}^* = \left(\sqrt{3} + 1, \ \sqrt{3} + 1 \right)$$

At the right end, we get

$$z = \frac{1}{2}\sqrt{2}, \qquad \mathbf{y}^* = \left(\frac{\sqrt{6}}{4}, \ \frac{\sqrt{2}}{4} \right), \qquad \mathbf{x}^* = \left(\sqrt{3} - 1, \ \sqrt{3} - 1 \right)$$

These are the same boundary functions derived in [8]. They are shown in Figure 1.

Figure 1. Boundary functions for D_2 with approximation order 1, at left and right ends.

7.2. Example 2: CL(3) Multiwavelet

The Chui–Lian multiwavelet CL(3) [16] has recursion coefficients

$$H_0 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 10 - 3\sqrt{10} & 5\sqrt{6} - 2\sqrt{15} \\ 5\sqrt{6} - 3\sqrt{15} & 5 - 3\sqrt{10} \end{pmatrix}, \qquad H_1 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 30 + 3\sqrt{10} & 5\sqrt{6} - 2\sqrt{15} \\ -5\sqrt{6} - 7\sqrt{15} & 15 - 3\sqrt{10} \end{pmatrix}$$

$$H_2 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 30 + 3\sqrt{10} & -5\sqrt{6} + 2\sqrt{15} \\ 5\sqrt{6} + 7\sqrt{15} & 15 - 3\sqrt{10} \end{pmatrix}, \qquad H_3 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 10 - 3\sqrt{10} & -5\sqrt{6} + 2\sqrt{15} \\ -5\sqrt{6} + 3\sqrt{15} & 5 - 3\sqrt{10} \end{pmatrix}$$

It has approximation order 3.

Here $\rho_0 = \rho_1 = 2$, so we need a vector of two boundary functions at each end. We can impose approximation order 2.

The original completion is

$$\Delta_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\sqrt{6}/8 - 7\sqrt{15}/40 & 3\sqrt{10}/40 - 3/8 & \sqrt{6}/8 - 3\sqrt{15}/40 & 3\sqrt{10}/40 - 1/8 \\ 0 & 0 & 3\sqrt{10}/40 - 3/8 & \sqrt{6}/8 + 7\sqrt{15}/40 & 1/8 - 3\sqrt{10}/40 & \sqrt{6}/8 - 3\sqrt{15}/40 \end{pmatrix}$$

After applying the algorithm in Section 6, we find that the top two rows of Q_L must be

$$Q_L = \begin{pmatrix} \sqrt{2}/2 & 0 & -\sqrt{30}/8 & -\sqrt{2}/8 \\ -\sqrt{24\sqrt{10}+1559}/\sqrt{3848} & \sqrt{2}/4 & -\sqrt{7041-36\sqrt{10}}/\sqrt{15392} & -\sqrt{191-60\sqrt{10}}/\sqrt{15392} \end{pmatrix}$$
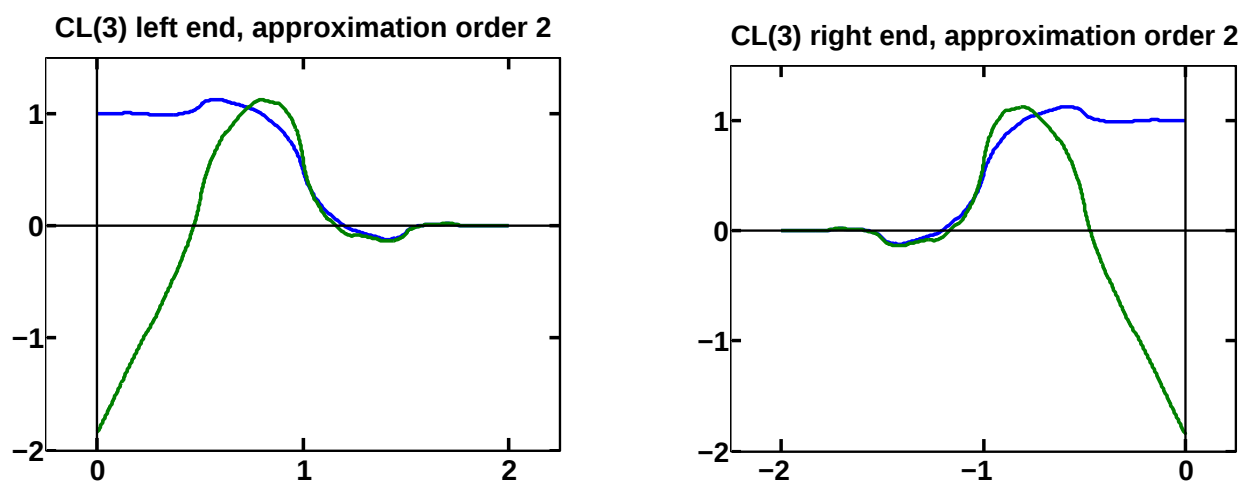
This leads to

$$A = \begin{pmatrix} 0.7071 & 0 \\ -0.6518 & 0.3536 \end{pmatrix}, \qquad B = \begin{pmatrix} 0.6980 & -0.0796 & 0.0091 & -0.0796 \\ 0.6613 & 0.0835 & -0.0095 & -0.0754 \end{pmatrix}$$

Only the numerical approximations of A, B are shown, since the exact expressions get very messy.

This is the same solution found in [8]. The boundary functions at the right end are the left end functions reversed, due to the symmetry of CL(3). The graphs are shown in Figure 2.

Figure 2. Boundary functions for CL(3) with approximation order 2, at left and right ends.

7.3. Example 3: DGHM Multiwavelet

The Donovan-Geronimo-Hardin-Massopust multiwavelet [17] has approximation order 2 with recursion coefficients

$$H_0 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 12 & 16\sqrt{2} \\ -\sqrt{2} & -6 \end{pmatrix}, \qquad H_1 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 12 & 0 \\ 9\sqrt{2} & 20 \end{pmatrix}$$

$$H_2 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 0 & 0 \\ 9\sqrt{2} & -6 \end{pmatrix}, \qquad H_3 = \frac{1}{20\sqrt{2}} \begin{pmatrix} 0 & 0 \\ -\sqrt{2} & 0 \end{pmatrix}$$

Its support is $[0, 2]$ instead of the expected $[0, 3]$, which causes some interesting effects.

We find $\rho_0 = 3$, $\rho_1 = 1$, so we have only a single boundary function at the left end, but three at the right end.

At the left end we can only enforce approximation order 1. The boundary scaling function found by our algorithm is the same one as in [8].
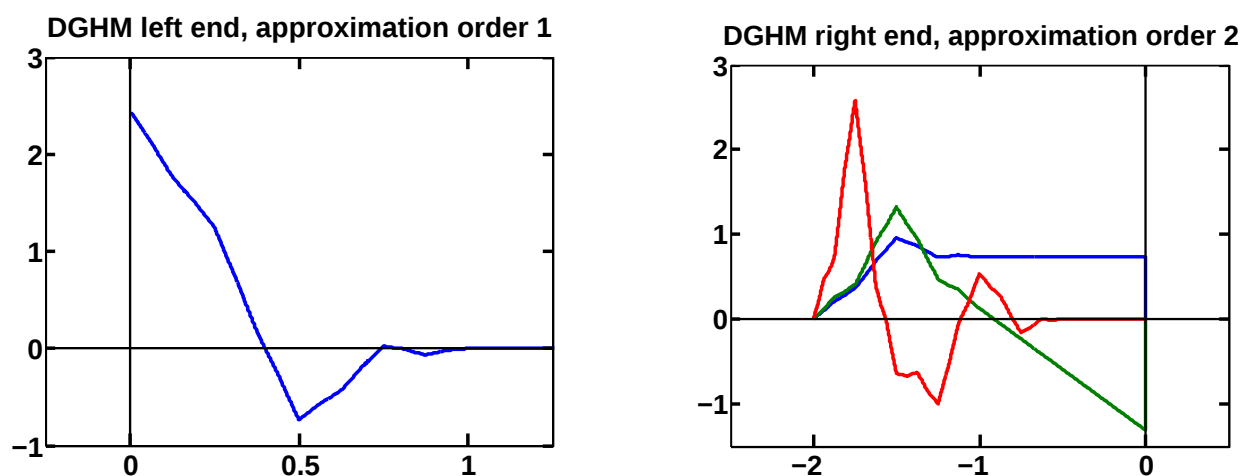
At the right end we can enforce approximation order 2. This determines two of the boundary functions, corresponding to the orthonormalized combination of the first two basic solutions from [8]. The third function (determined by the third row of Q_R) is arbitrary, except that the diagonal entry in Z should be smaller than $\sqrt{2}/2$ to ensure that the completion is regular.

As an example, we set the third column of Z to zero, and let the QR-factorization built into MATLAB choose an appropriate third row of Y. The result was

$$
Z = \begin{pmatrix} 0.7071 & 0 & 0 \\ -0.6250 & 0.3536 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \qquad
Y = \begin{pmatrix} 0.4477 & 0.3015 & 0.2345 & 0.3920 \\ 0.3279 & 0.0533 & 0.2902 & 0.5383 \\ -0.3711 & 0.2189 & 0.8638 & -0.2613 \end{pmatrix}
$$

The graphs are shown in Figure 3.

Figure 3. Boundary functions for DGHM. The single left boundary function has approximation order 1. Three right boundary functions provide approximation order 2.

8. Summary

Boundary functions for wavelets on a finite interval can be constructed from linear combinations of boundary-crossing internal functions, or by finding recursion coefficients, or by linear algebra techniques. The problem with the linear algebra approach is that it involves multiplication by arbitrary orthogonal matrices. A random choice leads to boundary recursion coefficients that produce an invertible transform but do not correspond to any actual boundary functions. Also, these coefficients do not provide any approximation orders.

In this paper, we present a linear algebra algorithm that works for all multiwavelets and removes most or all of the non-uniqueness while ensuring maximum possible boundary approximation orders.

References

1. Andersson, L.; Hall, N.; Jawerth, B.; Peters, G. Wavelets on Closed Subsets of the Real Line. In *Recent Advances in Wavelet Analysis*; Academic Press: Boston, MA, USA, 1994; Volume 3, pp. 1–61.

2. Chui, C.K.; Quak, E. Wavelets on a Bounded Interval. In *Numerical Methods in Approximation Theory*; Birkhäuser: Basel, 1992; Volume 105, pp. 53–75.

3. Cohen, A.; Daubechies, I.; Vial, P. Wavelets on the interval and fast wavelet transforms. *Appl. Comput. Harmon. Anal.* **1993**, *1*, 54–81.

4. Madych, W.R. Finite orthogonal transforms and multiresolution analyses on intervals. *J. Fourier Anal. Appl.* **1997**, *3*, 257–294.

5. Brislawn, C. Classification of nonexpansive symmetric extension transforms for multirate filter banks. *Appl. Comput. Harmon. Anal.* **1996**, *3*, 337–357.

6. Strang, G.; Nguyen, T. *Wavelets and Filter Banks*; Wellesley-Cambridge Press: Wellesley, MA, USA, 1996.

7. Williams, J.R.; Amaratunga, K. A discrete wavelet transform without edge effects using wavelet extrapolation. *J. Fourier Anal. Appl.* **1997**, *3*, 435–449.

8. Altürk, A.; Keinert, F. Regularity of boundary wavelets. *Appl. Comput. Harmon. Anal.* **2012**, *32*, 65–85.

9. Daubechies, I. Ten Lectures on Wavelets; In Proceeding of *CBMS-NSF Regional Conference Series in Applied Mathematics*; Society for Industrial and Applied Mathematics (SIAM): Philadelphia, PA, USA, 1992; Volume 61.

10. Keinert, F. Wavelets and Multiwavelets. In *Studies in Advanced Mathematics*; Chapman & Hall/CRC: Boca Raton, FL, USA, 2004.

11. Mallat, S.G. Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbf{R})$. *Trans. Am. Math. Soc.* **1989**, *315*, 69–87.

12. Strang, G. Wavelets and dilation equations: A brief introduction. *SIAM Rev.* **1989**, *31*, 614–627.

13. Gao, X.; Zhou, S. A study of orthogonal, balanced and symmetric multi-wavelets on the interval. *Sci. China Ser. F* **2005**, *48*, 761–781.

14. Lee, W.S.; Kassim, A.A. Signal and image approximation using interval wavelet transform. *IEEE Trans. Image Proc.* **2007**, *16*, 46–56.

15. Van Fleet, P.J. Multiwavelets and integer transforms. *J. Comput. Anal. Appl.* **2003**, *5*, 161–178.

16. Chui, C.K.; Lian, J.A. A study of orthonormal multi-wavelets. *Appl. Numer. Math.* **1996**, *20*, 273–298.

17. Donovan, G.C.; Geronimo, J.S.; Hardin, D.P.; Massopust, P.R. Construction of orthogonal wavelets using fractal interpolation functions. *SIAM J. Math. Anal.* **1996**, *27*, 1158–1192.