Multiresolution analysis based on quadratic Hermite interpolation - Part 1: Piecewise polynomial Curves

M. Dæhlen, T. Lyche
K. Mørken, and H-P. Seidel

Abstract. We study two simple multiresoultion analyses and their stability in the L^∞-norm: Faber decomposition and C^1 quadratic Hermite interpolation, both with nonuniform knot sequences. The use of the L^∞ norm is natural in many CAGD applications and it leads to schemes which are faster and simpler to implement than the wavelet schemes based on the L^2 norm. We have chosen to discuss quadratic Hermite interpolation because (i) it is a C^1 scheme with nice shape preserving properties, (ii) we have a certain sup norm stability in the wavelet spaces, (iii) there are local support bases for these spaces, (iv) the decomposition coefficients can be determined explicitly in real time, (v) it generalizes to splines over triangulations.

§1. Introduction

Multiresolution analysis (MRA) and wavelets ([4]) have turned out to be versatile tools both within Mathematics itself and in applications. Polynomial splines give rise to an important class of MRAs, and in this paper we are going to study two simple spline MRAs. One of them is based on linear splines and the other on quadratic splines. In a wavelet setting one usually assumes the knot spacing to be uniform, but in this paper we allow nonuniform knots. Another standard feature of wavelets is orthogonality, but here we will be content with direct sum decompositions. We also deviate from the standard by measuring stability in L^∞ rather than in L^2; this is more natural in many applications.

The MRA based on linear splines is included as a template for the quadratic MRA. It turns out that the linear MRA is stable in L^∞, uniformly for all knots. The quadratic Hermite MRA does not have this property. The main advantage of the quadratic MRA is of course that it is C^1, and elsewhere we have shown that it can be generalized to bivariate functions [6].

A first draft of this paper was prepared shortly after the last author visited Oslo in 1995, and the main results were presented at a conference at Oberwolfach that year.

§2. Faber Decomposition

In 1909 Faber [7] presented a hierachical representation of functions based on piecewise linear interpolation, see also [9]. In Faber's construction, a function f defined on the interval $[0,1]$ is decomposed into the sum

$$f = f_0 + \sum_{k=0}^{\infty} g_k, \quad g_k = f_{k+1} - f_k, \tag{2.1}$$

where f_k is the piecewise linear interpolant to f on the dyadic partition

$$\Delta_k = \{i2^{-k}\}_{i=0}^{2^k}. \tag{2.2}$$

Faber also gave an explicit formula for the "wavelets" g_k. In our notation it takes the form

$$g_k(x) = \sum_{j=1}^{2^k} d_{j,k} B_1(2^{k+1}x - 2j + 1), \quad k \geq 0, \tag{2.3}$$

where $B_1(x)$ is the piecewise linear function with $B_1(i) = \delta_{i,0}$ for all i, and

$$d_{j,k} = f\Big(\frac{2j-1}{2^{k+1}}\Big) - \frac{1}{2}\left(f\Big(\frac{2j-2}{2^{k+1}}\Big) + f\Big(\frac{2j}{2^{k+1}}\Big)\right). \tag{2.4}$$

This formula can be derived by observing that (2.3) and (2.1) lead to the relation

$$d_{j,k} = g_k(x_j) = f_{k+1}(x_j) - f_k(x_j), \quad \text{where} \quad x_j = \frac{2j-1}{2^{k+1}}.$$

Since x_j is an interpolation point for f_{k+1} that is midway between the two interpolation points $(j-1)/2^k$ and $j/2^k$ for f_k, we have

$$f_{k+1}(x_j) = f(x_j), \quad f_k(x_j) = \frac{1}{2}\left(f\Big(\frac{j-1}{2^k}\Big) + f\Big(\frac{j}{2^k}\Big)\right),$$

and (2.4) follows.

Faber was interested in this decomposition because it makes it quite simple to construct continuous nonsmooth functions. Faber decomposition has many other attractive features, particularly from a computational perspective. It is simple to implement, it is fast, and it can be generalized to triangles. In addition all functions used to represent f_k and g_k are dilates and translates of one simple function B_1. In essence, Faber decomposition yields decent compression with little effort.

§3. Multiresolution Analysis

Before we continue, let us take the time to spell out precisely the ingredients of a multiresolution analysis. Traditionally a multiresolution analysis consists of a nested sequence of subspaces of $L^2[0,1]$, see [4], but here we propose instead to use a nested sequence of subspaces of $C[0,1]$, equipped with the uniform norm, cf. [5]for a related construction. This gives us better control of the uniform norm of perturbation errors at the expense of losing the formalism of Hilbert spaces.

3.1 Basic ingredients

A Multiresolution Analysis consists of

(i) A collection $\{V_k\}_{k=0}^{\infty}$ of nested subspaces of $C[0,1]$ (with the uniform norm),

$$V_0 \subset V_1 \subset \cdots V_k \subset \cdots,$$

that are dense in $C[0,1]$,

$$\overline{\cup_{k=0}^{\infty} V_k} = C[0,1].$$

(ii) A collection $\{Q_k\}_{k=0}^{\infty}$ of uniformly bounded linear projectors from $C[0,1]$ onto V_k,

$$Q_k : C[0,1] \to V_k, \qquad \text{for } k = 0,\, 1,\, \ldots.$$

For each positive integer k we can then define the wavelet spaces

$$W_k = \{f \in V_{k+1} \mid Q_k f = 0\}, \tag{3.1}$$

which gives a decomposition of $C[0,1]$ as the direct sum

$$C[0,1] = V_0 + W_0 + W_1 + W_2 + \cdots. \tag{3.2}$$

In particular, each $f \in C[0,1]$ can be written as

$$f = f_0 + g_0 + g_1 + g_2 + \cdots, \tag{3.3}$$

where $g_k = f_{k+1} - f_k$ is in W_k for $k \geq 0$, and $f_k = Q_k f$ is in V_k. This follows since by construction,

$$V_{k+1} = V_k + W_k \quad \text{and} \quad V_k \cap W_k = \{0\}. \tag{3.4}$$

In addition we have

$$\|f - Q_k f\| \leq (1 + \|Q_k\|)\,\mathrm{dist}(f, V_k)$$

which by the uniform boundedness of the operators $\{Q_k\}_{k=0}^{\infty}$ means that

$$\lim_{k\to\infty} \|f - f_k\| = 0$$

(the norm used here and throughout the paper is the L^{∞}-norm).

3.2 Bases for V_k and W_k

For practical computations we need a basis $\{\phi_{j,k}\}_{j\in\mathcal{I}_k}$ for V_k and a basis $\{\psi_{j,k}\}_{j\in\mathcal{J}_k}$ for W_k. Here $\mathcal{I}_k$ and $\mathcal{J}_k$ are sets that index the basis functions $\{\phi_{j,k}\}_j$ and $\{\psi_{j,k}\}_j$ for each fixed k. Given an $f_{k+1} = \sum_{j\in\mathcal{I}_{k+1}} c_{j,k+1}\phi_{j,k+1}$ in V_{k+1}, we can then decompose it as $f_{k+1} = f_k + g_k$ with $f_k = \sum_{j\in\mathcal{I}_k} c_{j,k}\phi_{j,k}$ in V_k and $g_k = \sum_{j\in\mathcal{J}_k} d_{j,k}\psi_{j,k}$ in W_k. The algorithm for computing all the coefficients $d_{j,k}$ and $c_{j,k}$ from the $d_{j,k+1}$ is called the decomposition algorithm. This process can clearly be reversed, and this is the reconstruction algorithm. Together these algorithms constitute the Fast Wavelet Transform (FWT). Note that we will often collect the basis functions and coefficients of f_k and g_k together in vectors and write $f_k = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$ and $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k$.

3.3 Definition of stability

For efficient and accurate numerical computations it is important that the relation between a function and its coefficients in the wavelet basis is stable. The coarsest space V_0 is usually very simple and of low dimension so we concentrate on the wavelet spaces. The definition below therefore focuses on stability in the subspace

$$\tilde{C}[0,1] = \cup_{k=0}^{\infty} W_k,$$

i.e., all functions $f \in C[0,1]$ such that $Q_0 f = 0$. To measure the size of f we use the uniform norm, but we also need a norm to measure the size of the coefficients. If $f_n = \sum_{j\in\mathcal{J}_k, k=1}^{n} d_{j,k}\psi_{j,k}$ is a wavelet, we can use the vector max-norm

$$\delta_n = \|(d_{j,k})_{k=0, j\in\mathcal{J}_k}^{n-1}\|_\infty = \max_{j,k} |d_{j,k}|. \tag{3.5}$$

Definition 3.1. *The wavelets $\{\psi_{j,k}\}$ are said to form a weakly stable basis for $\tilde{C}[0,1]$ if there exist constants $K_{1,n}$ and $K_{2,n}$ such that*

$$K_{1,n}\delta_n \le \Big\| \sum_{\substack{k=0\\ j\in\mathcal{J}_k}}^{n-1} d_{j,k}\psi_{j,k} \Big\| \le K_{2,n}\delta_n, \tag{3.6}$$

where δ_n is given by (3.5), and $K_{1,n}$ and $K_{2,n}$ have at most polynomial growth in n.

Definition 3.1 provides a definition of stability based on the uniform norm, but the coefficient norm that is employed is rather coarse. The wavelet components $\{g_k\}$ may be very different and using $\max_{j,k} |d_{j,k}|$ as a measure of the size of all of them seems very crude. A more natural norm in many contexts would be $\|\boldsymbol{d}_0\| + \cdots + \|\boldsymbol{d}_{n-1}\|$. This leads to an alternative definition of stability.

Definition 3.2. *Let $f_n = f_0 + \sum_{k=0}^{n-1} g_k$ be a wavelet decomposition with $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k$. The wavelet basis is said to be stable if there are constants K_1 and K_2 such that*

$$K_1^{-1}\|(\boldsymbol{c}_0, \boldsymbol{d}_0, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_{n-1})\| \le \|f_n\|_\infty \le K_2\big(\|\boldsymbol{c}_0\| + \|\boldsymbol{d}_1\| + \cdots + \|\boldsymbol{d}_{n-1}\|\big). \tag{3.7}$$

The right-hand side of this stability estimate involves the norm of the coefficients of all the g_k, which is natural in most applications since they represent different frequency components of the underlying function. However, one may wonder why we have retained the coarser norm on the left-hand side of (3.7)? The main reason is that from the inequalities in (3.7) we can obtain a natural perturbation result. If $f = \sum_{k=0}^{n-1} g_k = \sum_{k=0}^{n-1} \boldsymbol{\psi}_k^T \boldsymbol{d}_k$ and $\tilde{f} = \sum_{k=0}^{n-1} \tilde{g}_k = \sum_{k=0}^{n-1} \boldsymbol{\psi}_k^T \tilde{\boldsymbol{d}}_k$ is a perturbation of f, then we have

$$\frac{\|f - \tilde{f}\|}{\|f\|} \le K_1 K_2 \Big(\frac{\|\boldsymbol{c}_0\|}{\delta} + \frac{\|\boldsymbol{d}_0\|}{\delta} + \cdots + \frac{\|\boldsymbol{d}_{n-1}\|}{\delta} \Big),$$

where $\delta = \max_{j,k} |d_{j,k}|$. In other words the relative error in f is bounded by the sum of the relative errors in the coefficients, multiplied by the factor $K_1 K_2$ which serves as a condition number. In a perturbation result of this kind it seems more natural and convenient to scale the errors in each of the coefficient vectors by the largest wavelet coefficient rather than the sum of all the norms of the wavelet coefficients.

§4. Nonuniform Faber decomposition

Faber decomposition can be generalized to nonuniform partitions. As above we have $\mathcal{F} = C[0,1]$ equipped with the L^∞-norm. For each nonnegative integer k we have $V_k = S_1(\Delta_k)$, the space of piecewise linear functions with breakpoints (knots) Δ_k. Here the knot vector is $\Delta_k = \boldsymbol{x}_k = (x_{j,k})_{j=-1}^{n_k+1}$ with $x_{-1,k} = x_{0,k} = 0$ and $x_{n_k,k} = x_{n_k+1,k} = 1$, so $\dim V_k = n_k + 1$. The knots are nested in that

$$x_{j,k} = x_{2j,k+1} \quad \text{and} \quad x_{j,k} < x_{2j+1,k+1} < x_{j+1,k}, \tag{4.1}$$

so $n_{k+1} = 2n_k$. The B-splines in V_k are $\{B_{j,\boldsymbol{x}_k}\}_{j=-1}^{n_k-1}$, but we rename these as $\boldsymbol{\phi}_k = (\phi_{j,k})_{j=0}^{n_k}$.

The projector $Q_k : C[0,1] \to V_k$ is the operator that assigns to each f in $C[0,1]$ the function in V_k which interpolates f at the (distinct) points in Δ_k. Since we have $\|Q_k\| = 1$ for all k, these projectors are uniformly bounded. The wavelet spaces $\{W_k\}_{k>0}$ are given by (3.1) and have dimension n_k, the number of knots added to $\boldsymbol{x}_k$ to get to $\boldsymbol{x}_{k+1}$. It is easy to see that a typical g_k in W_k can be expressed on the form

$$g_k = \sum_{j=1}^{n_k} d_{j,k}\psi_{j,k} = \boldsymbol{\psi}_k^T \boldsymbol{d}_k, \tag{4.2}$$

where $\psi_{j,k} = \phi_{2j-1,k+1}$ for $j = 1, \ldots, n_k$. These functions therefore form a basis for W_k. As in Section 2 we find

$$d_{j,k} = f(x_{2j-1,k+1}) - \big(\lambda_{j,k} f(x_{2j-2,k+1}) + (1-\lambda_{j,k}) f(x_{2j,k+1})\big), \tag{4.3}$$

where

$$\lambda_{j,k} = \frac{x_{j,k} - x_{2j-1,k+1}}{x_{j,k} - x_{j-1,k}}. \tag{4.4}$$

In view of (4.1) we see that $0 < \lambda_{j,k} < 1$ for all j, k.

The fundamental algorithms for dealing with the Faber decomposition are summarized below.

Reconstruction

Let $f_k = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$ be a function in V_k. This spline can be lifted into V_{k+1} as $f_k = \boldsymbol{\phi}_{k+1}^T \boldsymbol{c}_{k+1}$ via the simple formulas

$$\begin{aligned} c_{2j,k+1} &= c_{j,k}, && \text{for } j = 0, \ldots, n_k, \\ c_{2j+1,k+1} &= \lambda_{j,k} c_{j,k} + (1-\lambda_{j,k}) c_{j+1,k}, && \text{for } j = 0, \ldots, n_k - 1, \end{aligned}$$

with $\lambda_{j,k}$ given by (4.4).

Similarly, a function $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k$ in W_k is lifted into V_{k+1} as $g_k = \boldsymbol{\phi}_{k+1}^T \hat{\boldsymbol{c}}_{k+1}$ via the formulas

$$\begin{aligned} \hat{c}_{2j,k+1} &= 0, && \text{for } j = 0, \ldots, n_k, \\ \hat{c}_{2j+1,k+1} &= d_{j,k}, && \text{for } j = 0, \ldots, n_k - 1. \end{aligned}$$

Decomposition

Let $f_{k+1} = \boldsymbol{\phi}_{k+1}^T \boldsymbol{c}_{k+1}$ in V_{k+1} be given. Since $V_{k+1} = V_k + W_k$, we can split f into two components, one in V_k and one in W_k. The component in V_k is $f_k = Q_k f = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$, for which we find

$$c_{j,k} = c_{2j,k+1}, \qquad \text{for } j = 0, \ldots, n_k,$$

since every other breakpoint of f_{k+1} is also a breakpoint of f_k. The wavelet component g_k in W_k is given by $g_k(x) = \boldsymbol{\psi}_k^T \boldsymbol{d}_k = f_{k+1} - f_k$, and we find

$$d_{j,k} = c_{2j-1,k+1} - \big(\lambda_{j,k} c_{2j-2,k+1} + (1 - \lambda_{j,k}) c_{2j,k+1}\big) \tag{4.5}$$

with $\lambda_{j,k}$ given by (4.4). In the case of uniform knots this formula simplifies to (2.4).

Stability of Faber decomposition

An investigation into the stability of the Faber decomposition requires a bit more work. The first conclusion is that the wavelet basis is a weakly stable basis for $C[0,1]$.

Theorem 4.1. *Let $f_n = \sum_{k=0}^{n-1} g_k$ be a wavelet decomposition with $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k$ and $f_0 = 0$. Then*

$$\frac{1}{2} \max_{0 \le k \le n-1} \|\boldsymbol{d}_k\| \le \Big\| \sum_{k=0}^{n-1} \boldsymbol{\psi}_k^T \boldsymbol{d}_k \Big\| \le n \max_{0 \le k \le n-1} \|\boldsymbol{d}_k\|. \tag{4.6}$$

For a uniform partition the constant n in the rightmost inequality can be replaced by $2n/3 + 1$. In the uniform case the growth $2n/3$ is best possible in the sense that if $d_{j,k}$ is constant for all j and all k, then

$$\|f_n - f_0\|_{L^\infty[0,1]} \ge (2n/3) \max_{1 \le k \le n-1} \delta_k. \tag{4.7}$$

Proof: For convenience we set $f = f_n = \sum_{k=0}^{n-1} g_k$.

From (4.3) we obtain

$$|d_{j,k}| \le \big|f(x_{2j-1,k+1})\big| + \lambda_{j,k} \big|f(x_{2j-2,k+1})\big| + (1 - \lambda_{j,k}) \big|f(x_{2j,k+1})\big| \le 2\|f\|_\infty$$

which leads to the first inequality in (4.6). The second inequality in (4.6) follows from the triangle inequality and the fact that $\|g_k\| \le \|\boldsymbol{d}_k\|_\infty$,

$$\Big\| \sum_{k=0}^{n-1} g_k \Big\| \le \sum_{k=0}^{n-1} \|g_k\| \le \sum_{k=0}^{n-1} \|\boldsymbol{d}_k\|_\infty \le n \max_{0 \le k \le n-1} \|\boldsymbol{d}_k\|.$$

For the rest of the proof we work on uniform grids and recall that $f(0) = f(1) = 0$. The fundamental identity that we will use repeatedly is (2.4) which we now write as

$$f\big((2j-1)/2^{k+1}\big) = d_{j,k} + \frac{f\big((j-1)/2^k\big) + f\big(j/2^k\big)}{2}, \tag{4.8}$$

for $j = 0, 1, \ldots, 2^k - 1$.

To prove (4.7) we consider the function f^* for which $d_{j,k} = 1$ for all j and k. Our aim is to determine the maximum of f^* on each grid $\Delta_k = \{j2^{-k}\}_{j=0}^{2^k}$; let us denote this maximum by α_k. Suppose that the left hand side of (4.8) equals α_{k+1}. We observe that since one of j and $j+1$ must be odd and one even, one of the two function values on the right are computed on the grid Δ_k and one on a grid Δ_ℓ for some $\ell < k$ (the grid point belongs to Δ_i for $i = \ell$, $\ell+1$, …, k, but it does not belong to $\Delta_{\ell-1}$). Since the sequence (α_k) increases with k, the largest possible value for α_{k+1} is obtained if the two function values on the right correspond to α_k and α_{k-1}. We therefore have the recurrence relation

$$\alpha_{k+1} = 1 + (\alpha_k + \alpha_{k-1})/2 \tag{4.9}$$

which has the solution

$$\alpha_k = \frac{2}{3}k + \frac{2}{9} + \frac{1}{9}\left(-\frac{1}{2}\right)^{k-1}, \tag{4.10}$$

where we have used the inital conditions $\alpha_1 = 1$ and $\alpha_2 = 3/2$.

The number α_k given by (4.10) will be an overestimate of the growth of f_k^* unless there is some j such that $\alpha_{k+1} = f^*\big((2j+1)/2^{k+1}\big)$, and at the same time $\alpha_k = f^*(j/2^k)$ and $\alpha_{k-1} = f^*\big((j+1)/2^k\big)$ (possibly with α_k and α_{k-1} interchanged). Let us show that this does indeed happen.

On Δ_1 we have $\alpha_1 = f^*(1/2) = 1$, while on Δ_2 we have $\alpha_2 = f^*(1/4) = f^*(3/4) = 3/2$. It is therefore natural to guess that f^* attains its maximum on Δ_k at least two times, at the two points $a_k/2^k$ and $b_k/2^k$ where $a_1 = b_1 = 1$ and $a_2 = 1$ and $b_2 = 3$ and $a_k < b_k$ in general. To deduce a recurrence relation for a_k we note that if (4.9) is to hold, then there must be some j^* such that $a_{k+1} = 2j^* + 1$, and either $a_k = j^*$ and $2a_{k-1} = j^* + 1$ (if j^* is odd), or $a_k = j^* + 1$ and $2a_{k-1} = j^*$ (if j^* is even). In either case we have $a_{k+1} = a_k + 2a_{k-1}$. Combined with the inital conditions we find that

$$a_k = b_{k-1} = (2^k - (-1)^k)/3.$$

This proves that the norm of f^* on the grid Δ_k is α_k, from which the last claim follows. ■

Theorem 4.1 provides a stability estimate for Faber decomposition, but as mentioned above the coefficient norm that is employed is rather coarse. From the proof of Theorem 4.1 we can also deduce a stability result based on Definition 3.2.

Theorem 4.2. *Let $f_n = f_0 + \sum_{k=0}^{n-1} g_k$ be a wavelet decomposition with $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k$. Then*

$$\frac{1}{2}\|(\boldsymbol{c}_0, \boldsymbol{d}_0, \boldsymbol{d}_1, \dots, \boldsymbol{d}_{n-1})\| \le \|f_n\|_\infty \le \|\boldsymbol{d}_0\| + \|\boldsymbol{d}_1\| + \cdots + \|\boldsymbol{d}_{n-1}\|. \tag{4.11}$$

Note that the stability estimate in Theorem 4.2 is stricter than the one in Theorem 4.1 in that the estimate on the right in (4.11)is smaller than the one on the right in (4.6), while the estimate on the left remains the same. Theorem 4.2 shows that the condition number of Faber decomposition is 2.

§5. Multiresolution based on quadratic Hermite interpolation

A standard approximation technique is to interpolate position and first derivative at two points by a cubic polynomial. A similar approximation scheme can be constructed with piecewise quadratic polynomials, see [1], [2], [3] and [8].

We start by constructing a sequence of nested spaces of quadratic splines on the interval $[0,1]$. We assume that we have a sequence of knots $(\boldsymbol{x}_k)_{k=0}^{K}$ for quadratic splines on the interval $[0,1]$, i.e., the sequence $\boldsymbol{x}_k = (x_{j,k})_{j=-2}^{n_k+2}$ is ordered as

$$x_{-2,k} \le x_{-1,k} \le x_{0,k} < x_{1,k} < \cdots < x_{n_k-1,k} < x_{n_k,k} \le x_{n_k+1,k} \le x_{n_k+2,k}$$

with $[x_{0,k}, x_{n_k,k}] = [0,1]$ and $n_0 = 1$. The knots should be nested as in the linear case above in the sense that $\boldsymbol{x}_k$ is assumed to be obtained from $\boldsymbol{x}_{k-1}$ by inserting one new knot between each knot in $[0,1]$,

$$x_{2j,k} = x_{j,k-1}, \quad \text{and} \quad x_{j,k-1} < x_{2j+1,k} < x_{j+1,k-1} \quad \text{for } j = 0,\, 1,\, \ldots,\, n_{k-1} - 1.$$

From this it follows that n_k and n_{k-1} must be related by $n_k = 2n_{k-1}$. With these knots we can construct a nested sequence of spaces of quadratic splines

$$V_k = \operatorname{span}\{B_{j,\boldsymbol{x}_k}\}_{j=-2}^{n_k-1}, \qquad \dim V_k = n_k + 2,$$

where $B_{j,\boldsymbol{x}_k}$ is the quadratic B-spline with knots $(x_{j,k}, x_{j+1,k}, x_{j+2,k}, x_{j+3,k})$. For simplicity we introduce the notation $\phi_{j,k} = B_{j,\boldsymbol{x}_k}$.

For $k \ge 1$, we have an approximation operator Q_k that constructs quadratic splines in V_k by interpolating functions in $C^1[0,1]$ (functions on $[0,1]$ that are continuous and have a continuous derivative), at the knots in $\boldsymbol{x}_{k-1}$ that lie in $[0,1]$,

$$\left.\begin{aligned} Q_k f(x_{j,k-1}) &= f(x_{j,k-1}), \\ D(Q_k f)(x_{j,k-1} &= (Df)(x_{j,k-1}), \end{aligned}\right\} \quad \text{for } j = 0,\, 1,\, \ldots,\, n_{k-1}$$

(note that Q_K does not make sense for $k < 1$). If $Q_k f$ is written in terms of the B-splines in V_k as $Q_k f = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$ we find from this

$$\begin{aligned} c_{2j,k} &= f(x_{2j+2}) - Df(x_{2j+2})(x_{2j+2} - x_{2j+1})/2, \\ c_{2j+1,k} &= f(x_{2j+2}) + Df(x_{2j+2})(x_{2j+3} - x_{2j+2})/2, \end{aligned} \tag{5.1}$$

for $j = -1, \ldots, n_k/2 - 1$. Here the second subscript k has been omitted from all the x's. Note that these operators satisfy the important identity $Q_{k-1}Q_k = Q_{k-1}$; this can be exploited during wavelet decomposition and in obtaining stability results, se below.

Uniform boundedness of the operators

As we have seen above it is essential that the approximation operators $\{Q_k\}$ are uniformly bounded. This is indeed the case.

Lemma 5.1. *Let $\|\cdot\|$ denote the $L^\infty[0,1]$-norm and let $\|f\|^1 = \max\{\|f\|, \|Df\|\}$ be the norm in $C^1[0,1]$. For any $f \in C^1[0,1]$, the interpolant $Q_k f$ satisfies the inequalities*

$$\begin{aligned} \big|(Q_k f)(x)\big| &\leq \|f\| + h_0\|Df\|/2 \\ \big|D(Q_k f)(x)\big| &\leq 3\|Df\|, \end{aligned} \tag{5.2}$$

where $h_0 = \max_j(x_{j+1,0} - x_{j,0})$. In other words, the operator Q_k is bounded independently of k,

$$\|Q_k\|^1 \leq 3 + \frac{h_0}{2} \qquad \text{for all } k. \tag{5.3}$$

Proof: In this proof all second subscripts to c's and x's have been omitted, but are understood to be k. We write $Q_k f$ as $Q_k f = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$, with the coefficients given by (5.1). By standard properties of B-splines we have

$$\begin{aligned} \big|Q_k f(x)\big| &\leq \max_j |c_j|, \\ \big|D(Q_k f)(x)\big| &\leq \max_j 2|c_j - c_{j-1}|/(x_{j+2} - x_j). \end{aligned} \tag{5.4}$$

From (5.1) we conclude immediately that $|c_j| \leq \|f\| + h_0\|Df\|/2$. To bound the derivative we consider even and odd coefficients separately. We find from (5.1) that

$$2\frac{c_{2j+1} - c_{2j}}{x_{2j+3} - x_{2j+1}} = Df(x_{2j+2})$$

while

$$\begin{aligned} 2\frac{c_{2j} - c_{2j-1}}{x_{2j+2} - x_{2j}} = 2&\frac{f(x_{2j+2}) - f(x_{2j})}{x_{2j+2} - x_{2j}} - \\ &\frac{(x_{2j+2} - x_{2j+1})Df(x_{2j+2}) + (x_{2j+1} - x_{2j})Df(x_{2j})}{x_{2j+2} - x_{2j}}. \end{aligned}$$

From this the second bound in (5.2) follows. ■

As usual, the wavelet space W_k consists of the error functions that result when functions in V_{k+1} are approximated from V_k,

$$W_k = \{f \in V_{k+1} \mid Q_k f = 0\}, \qquad \dim W_k = \dim V_{k+1} - \dim V_k = n_k.$$

As in the linear case we have a basis for W_k consisting of some of the B-splines in V_{k+1}. Specifically, we have $W_k = \operatorname{span}\{\psi_{j,k}\}_{j=1}^{n_k}$ where

$$\left.\begin{aligned} \psi_{2j+1,k} &= B_{4j,k+1}, \\ \psi_{2j+2,k} &= B_{4j+1,k+1}, \end{aligned}\right\} \qquad \text{for } j = 0, \ldots, n_{k+1}/4 - 1.$$

These are all the B-splines in V_{k+1} that vanish at $x_{j,k-1}$ for $j = 0, \ldots, n_{k-1}$ (note that $n_{k+1}/4 = n_{k-1}$).

The reconstruction algorithm

Suppose that $f_k = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$ is a spline in V_k. Since $V_k \subseteq V_{k+1}$ it can also be written $f_k = \boldsymbol{\phi}_{k+1}^T \boldsymbol{c}_{k+1}$ where the coefficients $\boldsymbol{c}_{k+1}$ are given by

$$\begin{aligned} c_{2j-2,k+1} &= \frac{(x_{2j+2} - x_{2j-1})c_{j-2,k} + (x_{2j-1} - x_{2j-2})c_{j-1,k}}{x_{2j+2} - x_{2j-2}}, \\ c_{2j-1,k+1} &= \frac{(x_{2j+2} - x_{2j+1})c_{j-2,k} + (x_{2j+1} - x_{2j-2})c_{j-1,k}}{x_{2j+2} - x_{2j-2}}, \end{aligned} \tag{5.5}$$

for $j = 0, \ldots, n_k$ (the second subscript in all the x's is $k + 1$). Converting from a representation in W_k to a representation in V_{k+1} is simpler. If $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k = \boldsymbol{\phi}_{k+1}^T \tilde{\boldsymbol{c}}_{k+1}$ we have $\tilde{c}_{4j-2,k+1} = \tilde{c}_{4j-1,k+1} = 0$ for $j = 0, \ldots, n_{k+1}/4 - 1$ and

$$\left.\begin{aligned} \tilde{c}_{4j,k+1} &= d_{2j+1,k} \\ \tilde{c}_{4j+1,k+1} &= d_{2j+2,k} \end{aligned}\right\} \qquad \text{for } j = 0, \ldots, n_{k+1}/4 - 1. \tag{5.6}$$

The decomposition algorithm

A spline in $f_{k+1} = \boldsymbol{\phi}_{k+1}^T \boldsymbol{c}_{k+1}$ in V_{k+1} can be decomposed as $f_{k+1} = f_k + g_k$ where $f_k = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$ is in V_k and $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k$ is in W_k. The coefficients of f_k are obtained by solving the two equations (5.5) with respect to $c_{j-2,k}$ and $c_{j-1,k}$,

$$\begin{aligned} c_{2j-2,k} &= \frac{(x_{4j+1} - x_{4j-2})c_{4j-2,k+1} - (x_{4j-1} - x_{4j-2})c_{4j-1,k+1}}{x_{4j+1} - x_{4j-1}} \\ c_{2j-1,k} &= \frac{-(x_{4j+2} - x_{4j+1})c_{4j-2,k+1} + (x_{4j+2} - x_{4j-1})c_{4j-1,k+1}}{x_{4j+1} - x_{4j-1}} \end{aligned} \tag{5.7}$$

for $j = 0, \ldots, n_k/2$. Here the second subscript to the x's is $k + 1$, and we have replaced j by $2j$ since we have to compute the coefficients in pairs.

In order to write down the coefficients of $g_k = \boldsymbol{\psi}_k^T \boldsymbol{d}_k = f_{k+1} - f_k$ it is convenient to apply the reconstruction algorithm (5.5) and express f_k in terms of the B-spline basis in V_{k+1} as $f_k = \boldsymbol{\phi}_{k+1}^T \hat{\boldsymbol{c}}_{k+1}$. Then the coefficients $\boldsymbol{d}_k$ of g_k are given by

$$d_{2j+i+1,k} = c_{4j+i-1,k+1} - \hat{c}_{4j+i-1,k+1}, \qquad \text{for } i = 0, 1 \tag{5.8}$$

and $j = 0, \ldots, n_k/2 - 1$.

Stability

Before considering the stability of the proposed wavelet transform in detail, we need another identity for decomposition. The formulas in (5.7) project down from a spline in V_{k+1} to a spline in V_k. It is equally simple to project down from V_K directly into V_k. If $f_K = \boldsymbol{\phi}_K^T \boldsymbol{c}_K$, the approximation $f_k = \boldsymbol{\phi}_k^T \boldsymbol{c}_k$ in V_k is given by

$$\begin{aligned} c_{2j-2,k} &= \frac{(x_{J+1,K} - x_{2j-1,k})c_{J-2,K} - (x_{J-1,K} - x_{2j-1,k})c_{J-1,K}}{x_{J+1,K} - x_{J-1,K}} \\ c_{2j-1,k} &= \frac{-(x_{2j+1,k} - x_{J+1,K})c_{J-2,K} + (x_{2j+1,k} - x_{J-1,K})c_{J-1,K}}{x_{J+1,K} - x_{J-1,K}} \end{aligned} \tag{5.9}$$

for $j = 0, \ldots, n_k/2$, where $J = 2^{K-k+1}j$.

Wavelet decomposition and reconstruction is nothing but changes of bases between the two bases for V_K, namely $\boldsymbol{\phi}_K$ and $(\boldsymbol{\phi}_1, \boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_{K-1})$. It is therefore essential that these changes are stable so that the computations can be performed without encountering serious problems with rounding errors.

Lemma 5.2. *Let f be a function in V_K with the representations*

$$f = \boldsymbol{\phi}_K^T \boldsymbol{c}_K = \boldsymbol{\phi}_1^T \boldsymbol{c}_1 + \boldsymbol{\psi}_1^T \boldsymbol{d}_1 + \cdots + \boldsymbol{\psi}_{K-1}^T \boldsymbol{d}_{K-1}$$

in the two bases $\boldsymbol{\phi}_K$ and $(\boldsymbol{\phi}_1, \boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_{K-1})$ for V_K, and let $\|\cdot\|$ denote the vector max-norm. Then

$$\kappa_K^{-1} \big\|(\boldsymbol{c}_1, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_{K-1})\big\| \le \|\boldsymbol{c}_K\| \le \|\boldsymbol{c}_1\| + \|\boldsymbol{d}_1\| + \cdots + \|\boldsymbol{d}_{K-1}\|, \tag{5.10}$$

where

$$\kappa_K = 2\big(1 + \max_{1 \le k \le K-2} \{r_k + r_{k+1}\}\big) \tag{5.11}$$

and r_k is defined by

$$r_k = \max_{0 \le j \le n_k/2} \left\{ \frac{x_{J-1,K} - x_{2j-1,k}}{x_{J+1,K} - x_{J-1,K}}, \frac{x_{2j+1,k} - x_{J+1,K}}{x_{J+1,K} - x_{J-1,K}} \right\} \tag{5.12}$$

with $J = 2^{K-k+1} j$.

Proof: We start with the right-most inequality. Let R_{K-1} denote the matrix that represents reconstruction of B-spline coefficients from V_{K-1} to V_K, i.e., if $\tilde{f} = \boldsymbol{\phi}_{K-1}^T \tilde{\boldsymbol{c}}_{K-1} = \boldsymbol{\phi}_K^T \tilde{\boldsymbol{c}}_K$ we have $\tilde{\boldsymbol{c}}_K = \boldsymbol{R}_{K-1} \tilde{\boldsymbol{c}}_{K-1}$. But then $\boldsymbol{c}_K = \boldsymbol{R}_{K-1} \boldsymbol{c}_{K-1} + \ddot{\boldsymbol{d}}_{K-1}$ where $\ddot{\boldsymbol{d}}_{K-1}$ denotes the coefficients of g_{K-1} expressed in terms of $\boldsymbol{\phi}_K$ according to the reconstruction formulas (5.6), i.e., the coefficients $\boldsymbol{d}_{K-1}$ are augmented with a certain number of zeros. The reconstruction formulas (5.5) amount to taking weighted averages of neighbouring coefficients in $\boldsymbol{c}_{K-1}$ and therefore $\|\boldsymbol{R}_{K-1}\| = 1$ (in the ℓ^∞-norm for matrices). Hence we have

$$\|\boldsymbol{c}_K\| = \|\boldsymbol{R}_{K-1} \boldsymbol{c}_{K-1} + \ddot{\boldsymbol{d}}_{K-1}\| \le \|\boldsymbol{c}_{K-1}\| + \|\boldsymbol{d}_{K-1}\|.$$

Applying the same argument repeatedly to $\boldsymbol{c}_{K-1}$, $\boldsymbol{c}_{K-2}$, …, $\boldsymbol{c}_2$ leads to the right-most inequality in (5.10).

To prove the left inequality in (5.10) we make use of the fact that we can project straight from V_K to V_k with the formuals in (5.9), without going via the intermediate spaces. The formulas in (5.9) may be written as $\boldsymbol{c}_k = \boldsymbol{P}_k \boldsymbol{c}_K$ where $\boldsymbol{P}_k$ is a $(n_K + 2) \times (n_k + 2)$-matrix with two nonzero entries in each row that sum to one. We then have the bound

$$\|\boldsymbol{c}_k\| \le \|\boldsymbol{P}_k\| \|\boldsymbol{c}_K\|.$$

From (5.9) we find that

$$\|\boldsymbol{P}_k\| \le \max_{0 \le j \le n_k/2} \left\{ \frac{x_{J+1,K} + x_{J-1,K} - 2x_{2j-1,k}}{x_{J+1,K} - x_{J-1,K}}, \frac{2x_{2j+1,k} - x_{J+1,K} - x_{J-1,K}}{x_{J+1,K} - x_{J-1,K}} \right\},$$

where $J = 2^{K-k+1} j$. This may be simplified to

$$\|\boldsymbol{P}_k\| \le 1 + 2r_k \tag{5.13}$$

where r_k is given by (5.12). In particular we have

$$\|\boldsymbol{c}_1\| \le (1 + 2r_1)\|c_K\|. \tag{5.14}$$

To bound the wavelet coefficents we note that $\boldsymbol{d}_k$ may be expressed as

$$\ddot{\boldsymbol{d}}_k = \boldsymbol{c}_{k+1} - \boldsymbol{R}_k \boldsymbol{c}_k = \boldsymbol{P}_{k+1}\boldsymbol{c}_K - \boldsymbol{R}_K \boldsymbol{P}_k \boldsymbol{c}_K,$$

where $\ddot{\boldsymbol{d}}_k$ denotes the wavelet coefficients at level k with the zeros interspersed. From (5.13) we then obtain

$$\|\boldsymbol{d}_k\| \le \big(\|\boldsymbol{P}_{k+1}\| + \|\boldsymbol{P}_k\|\big)\|\boldsymbol{c}_K\| \le 2(1 + r_k + r_{k-1})\|\boldsymbol{c}_K\| \tag{5.15}$$

for $1 \le k \le K-2$ and

$$\|\boldsymbol{d}_{K-1}\| \le (2 + r_{K-1})\|\boldsymbol{c}_K\|. \tag{5.16}$$

The largest of the right-hand sides in (5.14)–(5.16) is κ_K which leads to the left inequality in (5.10). ∎

Lemma 5.2 is a statement about the stability of the wavelet algorithms. From this we get a result about the stability of the wavelet basis.

Lemma 5.3. *Let f be a function in V_K with the representation*

$$f = \boldsymbol{\phi}_1^T \boldsymbol{c}_1 + \boldsymbol{\psi}_1^T \boldsymbol{d}_1 + \cdots + \boldsymbol{\psi}_{K-1}^T \boldsymbol{d}_{K-1}$$

in the wavelet basis for V_K. Then

$$3^{-1}\kappa_K^{-1}\big\|(\boldsymbol{c}_1, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_{K-1})\big\| \le \|f\| \le \|\boldsymbol{c}_1\| + \|\boldsymbol{d}_1\| + \cdots + \|\boldsymbol{d}_{K-1}\|, \tag{5.17}$$

where κ_K is given by (5.11).

Proof: This result follows by combining the classical stability estimate for quadratic B-splines,

$$3^{-1}\|\boldsymbol{c}_K\| \le \|f\| \le \|\boldsymbol{c}_K\|,$$

with Lemma 5.2. ∎

This stability result is not particularly good since the condition number $3\kappa_K$ depends on the knots and can become very large. There is a possibility that the projection from V_K to V_k can be performed by more well conditioned formulas than (5.9), but this seems unlikely. In the special case of uniform knots we have $x_{j,k} = j2^{-k}$ and $r_k = 2^{K-k-1} - 1/2$; from (5.11) we then see that

$$3\kappa_K = 6(1 + 2^{K-2} - 1/2 + 2^{K-3} - 1/2) = 9 \cdot 2^{K-2}.$$

Decomposition in practice

In a wavelet environment, the initial operation to be performed on a function is normally wavelet decomposition. So given a function f in $C^1[0,1]$, we first pick a finest grid $\boldsymbol{x}_K$ and compute a quadratic spline approximation $f_K = \boldsymbol{\phi}_K^T \boldsymbol{c}_K$ using the formulas (5.1). We can then apply the decomposition algorithms (5.7) and (5.8) successively and decompose f_K as $f_K = f_1 + g_1 + \cdots + g_{K-1}$. However, because of the simplicity of our approximation scheme, there is another possibility. Instead of computing the B-spline coefficients of f_k for $k < K$ by successively applying (5.7), we can use the explicit formulas for the coefficients (5.1). If the values of f and Df on the finest grid are computed initially (this must be done even when the standard decomposition algorithms are used) and then stored, this approach will be faster than successive applications of (5.7) since the formulas (5.1) are arithmetically simpler than the formulas in (5.7).

In certain situations the explicit formulas (5.1) for the B-spline coefficients can be exploited to great advantage. Suppose for example that f and Df are very expensive to compute and we want a compressed representation of f with as few nonzero wavelet coefficients as possible. If we follow the traditional approach and start at a fine level we are very likely to compute many function and derivative values that are later discarded. However, the explicit formulas may allow us to compute the wavelet decomposition bottom-up instead of top-down. First we would have to compute the lowest level approximation to f on a grid $\boldsymbol{x}_1$. We would then have to decide in which areas this approximation is not satisfactory and decide where further sampling of f and its derivative is required. If f is reasonably smooth there should only be isolated areas where the approximation is not good enough. This means that many of the wavelet coefficients $(d_{j,1})$ can be set to zero so there is no need to compute the corresponding function and derivative values. The nonzero coefficients can then be computed by formula (5.8) where only the coefficients $c_{4j+i-1,2}$ will require new function evaluations. After this update of the approximation, its quality is assessed again and new wavelet coefficients added at level 2. The main challenge of this bottom-up approach to computing the wavelet decomposition is deciding where more information is needed, which is an inherent problem with adaptive computations.

Acknowledgments. Supported in part by the Research Council of Norway through project STP28402 and SINTEF.

References

1. Chui, Charles K. and Harvey Diamond, A general framework for local interpolation, Numer. Math. **58** (1991), 569–581.
2. Chui, Charles K. and Johan M. de Villiers, Applications of optimally local interpolation to constructing interpolatory approximants and compactly supported wavelets, CAT report 284, Texas A& M University, 1993.
3. Dahmen, W., T. N. T. Goodman, and C. A. Micchelli, Compactly supported fundamental functions for spline interpolation, Numer. Math. **52** (1988), 639–664.

4. Daubechies, I., Ten Lectures on Wavelets, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
5. DeVore, R. A., B. Jawerth, and B. Lucier, Surface compression, Comput. Aided Geom. Design **9** (1992), 219–239.
6. Dæhlen, M., T. Lyche, K. Mørken, R. Schneider, and H.-P. Seidel, Multiresolution analysis on triangles based on quadratic Hermite interpolation, J. Comput. Appl. Math. **119** (2000), 97–114.
7. Faber, G., Über stetige functionen, Mathematische Annalen **66** (1909), 81–94.
8. Villiers, J. M. de, A convergence result in nodal spline interpolation, J. Approx. Theory **74(3)** (1993), 266–279.
9. Yserentant, Harry, Hierarchical bases, in *ICIAM 91*, Robert E. O'Malley Jr. (ed.), SIAM, Philadelphia, 1991, 256–276.