Surface Compression Using A Space of C^1 Cubic Splines With A Hierarchical Basis

Don Hong[1)] and Larry L. Schumaker [2)]

Abstract. A method for compressing surfaces associated with C^1 cubic splines defined on triangulated quadrangulations is described. The method makes use of hierarchical bases, and does not require the construction of wavelets.

§1. Introduction

We consider surfaces which are defined as the graphs of real-valued functions defined on a domain $\Omega \subset \mathbb{R}^2$. In particular, we deal with C^1 cubic splines defined on triangulations obtained from convex quadrangulations by drawing both diagonals in each quadrilateral. The aim is to develop a compression scheme for such spline surfaces which does not require the construction of wavelets.

Our motivation for trying this approach is the fact that except for certain box spline spaces defined on very special partitions, it is very difficult to construct wavelets corresponding to bivariate splines of smoothness C^1 or greater, see Remarks 8.1 and 8.2. Indeed, even the case of C^0 linear splines on general triangulations is unexpectedly complicated, see [4,8,9] and references therein.

The key to our method is to work with C^1 cubic spline spaces which can be parameterized locally using the well-known FVS-macro-elements, see [1,10,11,17]. The algorithms are based on constructing hierarchical bases for certain nested sequences of such spline spaces. These hierarchical bases have been used [3] as tools for solving boundary-value problems.

The method is easy to implement and is computationally efficient since it is not necessary to keep track of basis functions, and does not require solving any systems of equations in either the decomposition or reconstruction phases. Test results show that it can achieve good approximations with high compression rates.

The paper is organized as follows. In Sect. 2 we briefly review the idea of hierarchical bases and discuss their usefulness for compression purposes. Sect. 3

[1)] Department of Mathematics, East Tennessee State University, Johnson City, TN 37614, USA, hong@math.vanderbilt.edu.

[2)] Center for Constructive Approximation, Department of Mathematics, Vanderbilt University, Nashville, TN 37240, USA, s@mars.cas.vanderbilt.edu. Supported in part by the Army Research Office under grant DAAD 190210059.

reviews basic facts about C^1 cubic spline spaces on triangulated quadrangulations, while Sect. 4 describes the refinement process used to create sequences of nested spline spaces. Sect. 5 is devoted to the construction of hierarchical bases for the resulting spline spaces, while Sect. 6 goes into the details of the compression algorithm. Numerical examples can be found in Sect. 7. We conclude the paper with remarks and references.

§2. Hierarchical Bases

In this section we briefly review the idea of hierarchical bases and discuss their usefulness for compression purposes. Suppose

$$\mathcal{S}_0 \subset \mathcal{S}_1 \subset \mathcal{S}_2 \subset \cdots \subset \mathcal{S}_\ell$$

is a nested sequence of finite dimensional spaces of real-valued functions. Then a set of functions

$$\mathcal{B} := \bigcup_{k=0}^{\ell} \{B_i^k\}_{i=1}^{n_k}$$

is said to be a hierarchical basis for $\mathcal{S}_\ell$ provided

$$\mathcal{B}_m := \bigcup_{k=0}^{m} \{B_i^k\}_{i=1}^{n_k}$$

is a basis for $\mathcal{S}_m$ for each $m = 0, 1, \ldots, \ell$. Then every $s \in \mathcal{S}_\ell$ can be written in the form

$$s = \sum_{m=0}^{\ell} \sum_{i=1}^{n_m} c_i^m B_i^m, \tag{2.1}$$

and the partial sums

$$s_k := \sum_{m=0}^{k} \sum_{i=1}^{n_m} c_i^m B_i^m \tag{2.2}$$

are splines in the spaces $\mathcal{S}_k$ for each $0 \le k \le \ell$. The expansion (2.1) is particularly useful when

$$\|s - s_0\| > \|s - s_1\| > \cdots > \|s - s_{\ell-1}\|, \tag{2.3}$$

since in this case the sequence of splines $s_0, s_1, \ldots, s_\ell$ can be regarded as better and better approximations of s. Then if we only need an approximation to s, it is enough to know only part of the coefficients. For example, for the coarsest approximation s_0, it suffices to know only the coefficients $\{c_i^0\}_{i=1}^{n_0}$. Thus, in this case the representation (2.1) is well-suited for progressive transmission of coefficients.

The expansion (2.1) can also be used for compression provided that the basis functions are stable in the sense that small changes in the size of coefficients in

(2.1) lead to small changes in the size of $\|s\|$. To describe a compressed approximation of s, we can store (or transmit) only coefficients which are larger than some prescribed threshold. The ratio of retained coefficients to original coefficients will then describe the compression rate. (This ratio will not correspond to the true compression rate, since we would still need some way of describing which coefficients have been retained, but this can be done with standard coding techniques, see Remark 8.4).

§3. C^1 Cubic Splines on a Triangulated Quadrangulation

Suppose $\mathcal{V} := \{v_i\}_{i=1}^n$ is a set of points in $\mathbb{R}^2$. Then (cf. [11,12]), a set $\diamondsuit$ of quadrilaterals with vertices $\mathcal{V}$ is called a quadrangulation of Ω provided 1) Ω is the union of the quadrilaterals in $\diamondsuit$, and 2) the intersection of any two quadrilaterals in $\diamondsuit$ is either empty, a common vertex, or a common edge. We focus on quadrangulations where the largest angle in any quadrilateral is less than π. Given such a quadrangulation, let $\oplus$ be the triangulation which is obtained by drawing in both diagonals in each quadrilateral. We write $\mathcal{E}$ for the set of edges of $\diamondsuit$, where we assume each edge e has been assigned a specific orientation. Associated with $\oplus$, let

$$\mathcal{S}_3^1(\oplus) := \{s \in C^1(\Omega) : s|_T \in \mathcal{P}_3, \quad \text{all } T \in \oplus\},$$

be the corresponding space of C^1 cubic splines, where $\mathcal{P}_3$ is the space of cubic bivariate polynomials.

It is well known (cf. [11]) that

$$n := \dim \mathcal{S}_3^1(\oplus) = 3V + E,$$

where V and E are the number of vertices and edges of $\diamondsuit$, respectively. We now describe a basis for $\mathcal{S}_3^1(\oplus)$. For each $v \in \mathcal{V}$, let λ_v, λ_v^x and λ_v^y be the point-evaluation functionals defined on the space $C^1(\Omega)$ by

$$\lambda_v s = s(v), \qquad \lambda_v^x s = D_x s(v), \qquad \lambda_v^y s = D_y s(v). \tag{3.1}$$

For each oriented edge $e \in \mathcal{E}$, let γ_e be the linear functional such that

$$\gamma_e s = D_e s(u_e), \tag{3.2}$$

where u_e is the center of e and D_e is the directional derivative associated with a unit vector r_e which is perpendicular to e. Then it is well-known (cf. [10,11,17]) that the set of linear functionals

$$\Lambda := \{\lambda_i\}_{i=1}^n := \bigcup_{v\in\mathcal{V}} \{\lambda_v, \lambda_v^x, \lambda_v^y\} \cup \bigcup_{e\in\mathcal{E}} \gamma_e$$

is a minimal determining set for $\mathcal{S}_3^1(\oplus)$, i.e., each spline $s \in \mathcal{S}_3^1(\oplus)$ is uniquely determined by the values $\{\lambda_i s\}_{i=1}^n$. This can also be stated as follows:

Lemma 3.1. *Given any function $f \in C^1(\Omega)$, there is a unique spline $s_f \in \mathcal{S}_3^1(\diamondplus)$ satisfying*

1) $s_f(v) = f(v)$ *for all* $v \in \mathcal{V}$,
2) $D_x s_f(v) = D_x f(v)$ *and* $D_y s_f(v) = D_y f(v)$ *for all* $v \in \mathcal{V}$,
3) $D_e s_f(u_e) = D_e f(u_e)$ *for all* $e \in \mathcal{E}$.

The fact that Λ is a minimal determining set for $\mathcal{S}_3^1(\diamondplus)$ means that to store a given spline $s \in \mathcal{S}_3^1(\diamondplus)$ in a computer, we need only store the n-vector $(\lambda_1 s, \ldots, \lambda_n s)$. The entries of this vector are just values of s or its first derivatives at certain points. The process of evaluating s at any given point is greatly simplified by the fact that the above data actually determines s locally. More precisely, if Q is a quadrilateral of $\diamondsuit$, then s is uniquely determined on Q by the values $s(v), D_x s(v), D_y s(v)$ at the four vertices of Q, coupled with the values of $D_e s(u_e)$ for the four edges of Q. For especially efficient evaluation on Q, these 16 pieces of data can be used to compute the corresponding Bézier net for s, after which the standard de Casteljau algorithm can be applied to find values or derivatives of s (see [3,11]).

The following error bound for the Hermite interpolating spline can be established by standard methods using the Bramble-Hilbert lemma. Let h be the diameter of the largest triangle in $\diamondplus$, and let $W_p^m(\Omega)$ be the usual Sobolev space with semi-norm $|f|_{m,p}$.

Lemma 3.2. *Fix $2 \le m \le 4$ and $1 \le p \le \infty$. Then there exists a constant C depending only on m and the smallest angle in $\diamondplus$ such that*

$$\|D_x^\nu D_y^\mu (f - s_f)\|_p \le C h^{m-\nu-\mu} |f|_{m,p},$$

for every $f \in W_p^m(\Omega)$ and all $0 \le \nu + \mu \le m$.

§4. A Refinement Scheme

In this section we discuss a natural scheme for refining a given quadrangulation $\diamondsuit_0$ and its associated triangulation $\diamondplus_0$ to produce nested sequences

$$\diamondsuit_0 \subset \diamondsuit_1 \subset \diamondsuit_2 \subset \cdots \subset \diamondsuit_\ell \tag{4.1}$$

and

$$\diamondplus_0 \subset \diamondplus_1 \subset \diamondplus_2 \subset \cdots \subset \diamondplus_\ell. \tag{4.2}$$

We will use these in the next section to define nested sequences of cubic spline spaces.

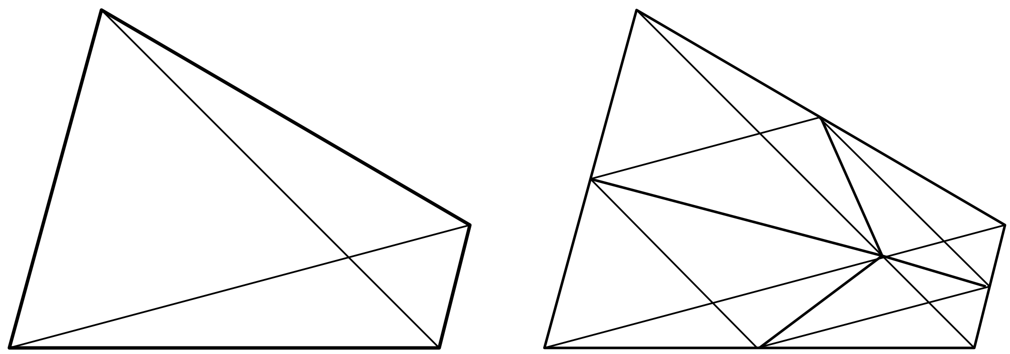


Fig. 1. One step of the refinement algorithm on a single quadrilateral.

Algorithm 4.1. *Let $\Diamond\!\!\!\!\!\mid_0$ be the triangulation associated with a quadrangulation $\Diamond_0$. For each Q in $\Diamond_0$,*

1) *let v_Q be the point where the two diagonals of Q intersect,*
2) *connect the point v_Q to the centers $w_{Q,1}, \ldots, w_{Q,4}$ of the edges of Q,*
3) *connect $w_{Q,i}$ to $w_{Q,i+1}$ for $i = 1, \ldots, 4$, where we identify $w_{Q,5} := w_{Q,1}$.*

It is clear that Algorithm 4.1 splits each quadrilateral in $\Diamond_0$ into four subquadrilaterals and each triangle in $\Diamond\!\!\!\!\!\mid_0$ into four subtriangles, see Fig. 1. This process can be repeated as often as desired to produce the nested sequences in (4.1) and (4.2). Let $\mathcal{V}_0$ be the set of vertices of the initial quadrangulation $\Diamond_0$. We write $\mathcal{V}_0^c$ for the set of points at intersections of diagonals arising in step 1 of the algorithm, and $\mathcal{V}_0^e$ for the set of points at midpoints of edges arising in step 2 of the algorithm. Then applying the algorithm repeatedly, we get analogous sets of points $\mathcal{V}_{m-1}^c$ and $\mathcal{V}_{m-1}^e$, and it is easy to see that the set of vertices of $\Diamond_m$ is just

$$\mathcal{V}_m := \mathcal{V}_{m-1} \cup \mathcal{V}_{m-1}^c \cup \mathcal{V}_{m-1}^e$$

for all $1 \leq m \leq \ell$.

Let V_m, E_m, and Q_m denote the number of vertices, edges, and quadrilaterals in the quadrangulation $\Diamond_m$ obtained after performing m steps of Algorithm 4.1 on an initial quadrangulation $\Diamond_0$.

Lemma 4.2. *For all $m \geq 0$,*

$$\begin{aligned} Q_m &= 4^m Q_0, \\ E_m &= 2^m E_0 + 2(4^m - 2^m) Q_0, \\ V_m &= V_0 + (2^m - 1) E_0 + (4^m - 2^{m+1} + 1) Q_0. \end{aligned}$$

Proof: The first formula follows immediately from $Q_m = 4Q_{m-1}$. For E_m we have the difference equation $E_m = 2E_{m-1} + 4Q_{m-1}$, and solving it leads to the stated formula for E_m. Finally, to get the third formula we solve the difference equation $V_m = V_{m-1} + E_{m-1} + Q_{m-1}$. $\square$

For comparison purposes, in Table 1 we give the numbers Q_m, E_m and V_m for $m = 0, \ldots, 8$, assuming that we start with a single quadrilateral. The table also shows the dimension d_m of $\mathcal{S}_3^1(\Diamond_m)$. We conclude this section with a result on the stability of the refinement process. Let θ_m be the smallest angle in the triangulation $\diamondplus_m$.

m	Q_m	E_m	V_m	d_m
0	1	4	4	16
1	4	12	9	39
2	16	40	25	115
3	64	144	81	387
4	256	544	289	1411
5	1024	2112	1089	5379
6	4096	8320	4225	20995
7	16384	33024	16641	82947
8	65536	131584	66049	329731

Tab. 1. The combinatorics of $\mathcal{S}_3^1(\diamondplus_m)$ for $m = 0, \ldots, 8$.

Theorem 4.3. *There exists a constant $0 < \kappa < 1$ depending only on θ_0 such that*

$$\theta_m \geq \kappa\theta_0, \qquad \textit{all } m > 0. \tag{4.3}$$

Proof: For a proof of (4.3) for $m = 1$, see Remark 8.6. For $m > 1$ the result follows from the observation (see the proof of Proposition 5.2 in [3]) that $\theta_m = \theta_1$ for all $m > 1$. $\square$

§5. A Nested Sequence of C^1 Cubic Splines

In this section we work with the nested sequence of C^1 cubic spline spaces

$$\mathcal{S}_3^1(\diamondplus_0) \subset \mathcal{S}_3^1(\diamondplus_1) \subset \cdots \subset \mathcal{S}_3^1(\diamondplus_\ell)$$

corresponding to (4.1) and (4.2). Our aim is to describe a hierarchical basis for $\mathcal{S}_3^1(\diamondplus_\ell)$.

For each $0 \leq m \leq \ell$, let $\mathcal{V}_m$ and $\mathcal{E}_m$ denote the sets of vertices and (oriented) edges of the m-th quadrangulation $\Diamond_m$ in the nested sequence (4.1). As in Sect. 4, let $\mathcal{V}_m^c$ be the set of diagonal crossing points of the quadrilaterals of $\Diamond_m$. For each vertex v of $\Diamond_\ell$, let $\lambda_v, \lambda_v^x, \lambda_v^y$ be the linear functionals defined in (3.1). If e is any edge of a quadrilateral, we write u_e for its midpoint. For each edge e of a quadrilateral, let γ_e be the linear functional defined in (3.2), and let $\tilde{\gamma}_e$ be the linear functional defined by

$$\tilde{\gamma}_e s := \widetilde{D}_e s(u_e),$$

where $\widetilde{D}_e$ is the directional derivative associated with the unit vector pointing in the direction of e.

We now construct a special minimal determining set for $\mathcal{S}_3^1(\diamondplus_\ell)$. Let

$$\Lambda_0 := \{\lambda_i^0\}_{i=1}^{n_0} := \bigcup_{v\in\mathcal{V}_0} \{\lambda_v, \lambda_v^x, \lambda_v^y\} \cup \bigcup_{e\in\mathcal{E}_0} \gamma_e$$

and for each $1 \le k \le \ell$, set

$$\Gamma_k := \{\lambda_i^k\}_{i=1}^{n_k} := \bigcup_{v\in\mathcal{V}_{k-1}^c} \{\lambda_v, \lambda_v^x, \lambda_v^y\} \cup \bigcup_{e\in\mathcal{E}_{k-1}} \{\lambda_{u_e}, \tilde{\gamma}_e\} \cup \bigcup_{e\in\mathcal{E}_k} \gamma_e.$$

Theorem 5.1. *For each $0 \le m \le \ell$, the set of linear functionals*

$$\Lambda_m := \Lambda_0 \cup \bigcup_{k=1}^{m} \Gamma_k$$

forms a minimal determining set for $\mathcal{S}_3^1(\diamondplus_m)$.

Proof: It is easy to see that setting the values $\{\lambda s\}_{\lambda\in\Lambda_m}$ is equivalent to setting

$$\bigcup_{v\in\mathcal{V}_m} \{\lambda_v s, \lambda_v^x s, \lambda_v^y s\} \cup \bigcup_{e\in\mathcal{E}_m} \gamma_e s. \tag{5.1}$$

By the results of Sect. 3, these values uniquely determine a spline $s \in \mathcal{S}_3^1(\diamondplus_m)$. $\square$

We now construct a hierarchial basis for $\mathcal{S}_3^1(\diamondplus_\ell)$. For each $1 \le i \le n_0$, let B_i^0 be the unique spline in $\mathcal{S}_3^1(\diamondplus_0)$ such that

$$\lambda_j^0 B_i^0 = \delta_{ij}, \qquad j = 1, \ldots, n_0. \tag{5.2}$$

In addition, for each $1 \le m \le \ell$ and each $1 \le i \le n_m$, let B_i^m be the unique spline in $\mathcal{S}_3^1(\diamondplus_m)$ such that

$$\begin{aligned} \lambda_j^m B_i^m &= \delta_{ij}, \qquad j = 1, \ldots, n_m, \\ \lambda_j^k B_i^m &= 0, \qquad j = 1, \ldots, n_k, \quad k = 1, \ldots, m-1. \end{aligned} \tag{5.3}$$

Theorem 5.2. *For each $0 \le m \le \ell$, the set of splines*

$$\mathcal{B}_m := \bigcup_{k=0}^{m} \bigcup_{i=1}^{n_k} \{B_i^k\}$$

forms a basis for $\mathcal{S}_3^1(\diamondplus_m)$.

Proof: By construction, the splines in $\mathcal{B}_m$ lie in $\mathcal{S}_3^1(\diamondplus_m)$ and are linearly independent. Then the result follows from the fact that the cardinality of $\mathcal{B}_m$ is equal to

$n_0 + n_1 + \cdots + n_m$, which is the cardinality of the minimal determining set Λ_m for $\mathcal{S}_3^1(\diamondplus_m)$. $\square$

Theorem 5.2 shows that $\mathcal{B}_\ell$ is a hierarchical basis for $\mathcal{S}_3^1(\diamondplus_\ell)$, and thus every spline $s \in \mathcal{S}_3^1(\diamondplus_\ell)$ has a unique hierarchical representation

$$s = \sum_{m=0}^{\ell} \sum_{i=1}^{n_m} c_i^m B_i^m. \tag{5.4}$$

We now show that the basis functions in Theorem 5.2 are local and stable. To make this more precise, given any vertex v of quadrangulation $\Diamond_m$, let $\mathrm{star}_m(v)$ be the union of the (at most four) quadrilaterals of $\Diamond_m$ which share the vertex v. Similarly, if u_e is the midpoint of some edge of $\Diamond_m$, let $\mathrm{nhb}_m(e)$ be the union of the (at most two) quadrilaterals of $\Diamond_m$ which share the edge e. Let $\Lambda_m^{(0)}$ and $\Lambda_m^{(1)}$ be the subsets of those linear functionals in Λ_m which involve function evaluation and derivative evaluation, respectively.

Theorem 5.3. *For each $0 \le m \le \ell$, the supports and sizes of B_i^m satisfy*

1) *$\operatorname{supp} B_i^m \subseteq \mathrm{star}_m(v)$ if λ_i^m involves evaluation at a vertex v of $\Diamond_m$,*
2) *$\operatorname{supp} B_i^m \subseteq \mathrm{nhb}_m(e)$ if λ_i^m involves evaluation at u_e for some edge e of $\Diamond_m$,*
3) *$\|B_i^m\|_\infty \le 1$ if $\lambda_i^m \in \Lambda_m^{(0)}$,*
4) *$\|B_i^m\|_\infty \le H_{m,i}$ if $\lambda_i^m \in \Lambda_m^{(1)}$, where $H_{m,i}$ is the maximal diameter of the triangles contained in $\operatorname{supp} B_i^m$.*

Proof: The claim about the supports of the B_i^m follows immediately from the fact (cf. the discussion in Sect. 3) that on each quadrilateral Q of $\Diamond_m$, a spline is determined from values at the four vertices of Q and at the midpoints of the four sides of Q. Now concerning the sizes of these basis functions, in case 3), it is easy to see that the Bézier coefficients of B_i^m on any subtriangles of $\diamondplus_m$ are bounded by 1. This implies $\|B_i^m\|_\infty \le 1$. When λ_i^m corresponds to a derivative, the Bézier coefficients on any subtriangle T of $\diamondplus_m$ are bounded by the diameter of T, and $\|B_i^m\|_\infty \le H_{m,i}$ follows. $\square$

Properties 1) and 2) of Theorem 5.3 insure that the basis functions in (5.4) are local. Combining these with properties 3) and 4) of the basis, we can now show that it is also stable in the sense that if s has small coefficients, then $\|s\|_\infty$ is also small.

Theorem 5.4. *Suppose $s \in \mathcal{S}_3^1(\diamondplus_\ell)$ is a spline whose coefficients satisfy*

$$|\lambda_i^m s| \le \begin{cases} \frac{\varepsilon}{16\ell}, & \text{if } \lambda_i^m \in \Lambda_m^{(0)}, \\ \frac{\varepsilon}{16\ell H_m}, & \text{if } \lambda_i^m \in \Lambda_m^{(1)}, \end{cases}$$

where H_m is the maximum of the $H_{m,i}$ appearing in Theorem 5.3. Then $\|s\|_\infty < \varepsilon$.

Proof: By the support properties of the basis, it follows that for any quadrilateral $Q \in \Diamond_\ell$, at most 16ℓ basis functions have support containing Q. The claim now follows from statements 3) and 4) of Theorem 5.3. $\square$

§6. Compression

In view of the discussion in Sect. 3, a spline $s \in \mathcal{S}_3^1(\diamondplus_\ell)$ is uniquely determined by the values

$$\bigcup_{v \in \mathcal{V}_\ell} \{s(v), D_x s(v), D_y s(v)\} \cup \bigcup_{e \in \mathcal{E}_\ell} D_e s(u_e). \tag{6.1}$$

By the results of the previous section, s is also uniquely determined by the coefficients appearing in the expansion (5.4). As we shall see below, generally many of these coefficients will be small, and we can replace them by zero to define a spline $\hat{s}$ which has fewer nonzero coefficients but is still close to s. This is the basis of our compression method.

In analogy with standard wavelet terminology, we refer to the process of computing the coefficients in (5.4) from the values (6.1) as decomposition, and the reverse process of computing the values (6.1) from the coefficients as reconstruction. The following theorem is the basis for a decomposition algorithm.

Theorem 6.1. *The coefficients in (5.4) are given by*

$$c_i^0 = \lambda_i^0 s, \qquad i = 1, \ldots, n_0 \tag{6.2}$$

and

$$c_i^m = \lambda_i^m (s - s_{m-1}), \qquad i = 1, \ldots, n_m, \quad m = 1, \ldots, \ell, \tag{6.3}$$

where

$$s_{m-1} := \sum_{k=0}^{m-1} \sum_{i=1}^{n_k} c_i^k B_i^k. \tag{6.4}$$

Proof: The claim follows immediately from the duality properties (5.2) and (5.3) of the hierarchical basis. $\square$

Theorem 6.1 can easily be turned into an algorithm for computing the coefficients in (5.4).

Algorithm 6.2. (Decomposition)

1) *Use (6.2) to compute* $\{c_i^0\}_{i=1}^{n_0}$ *from* $\{\lambda_v s, \lambda_v^x s, \lambda_v^y s\}_{v \in \mathcal{V}_0} \cup \{\gamma_e s\}_{e \in \mathcal{E}_0}$.
2) *For* $m = 1, \ldots, \ell$,
   a) *Form the spline* s_{m-1} *as in (6.4)*
   b) *Compute* $\{c_i^m\}_{i=1}^{n_m}$ *as in (6.3).*

For the purposes of compression, we now apply Theorem 5.3 to describe a thresholding strategy.

Algorithm 6.3. (Thresholding)

1) *Choose* ε.
2) *For each* $m = 1, \ldots, \ell$,
   a) *Drop the coefficient corresponding to* $\lambda_i^m \in \Lambda_m^{(0)}$ *if it is smaller than* ε.
   b) *Drop the coefficient corresponding to* $\lambda_i^m \in \Lambda_m^{(1)}$ *if it is smaller than* $2^m\varepsilon$.

The decomposition algorithm will give good compression rates when the expansion (5.4) contains many small coefficients. In view of (6.3), the size of the coefficients c_i^m depend on the size of $s - s_{m-1}$ and its first derivatives. In this connection we have the following result.

Theorem 6.4. *Given* $f \in W_p^4(\Omega)$ *with* $1 \le p \le \infty$, *suppose* $s \in \mathcal{S}_3^1(\diamondsuit)$ *satisfies* $\lambda_i^\ell s = \lambda_i^\ell f$, $i = 1, \ldots, n_\ell$. *Then for all* $1 \le m \le \ell$,

$$\|s - s_{m-1}\|_p \le C_1 h_{m-1}^4 |f|_{4,p}. \tag{6.5}$$

Moreover, for any unit vector u,

$$\|D_u(s - s_{m-1})\|_p \le C_2 h_{m-1}^3 |f|_{4,p}, \tag{6.6}$$

where D_u *is the directional derivative corresponding to* u. *Here* h_{m-1} *is the mesh size of* $\diamondsuit_{m-1}$, *i.e., the diameter of the largest triangle in* $\diamondsuit_{m-1}$. *The constants* C_1 *and* C_2 *depend only on* ℓ *and the smallest angle* θ_0 *in* $\diamondsuit_0$.

Proof: By Lemma 3.2,

$$\|f - s_k\|_p \le C h_k^4 |f|_{4,p},$$

for all $1 \le k \le \ell$. Then (6.5) follows with $C_1 = 2C$ from the triangle inequality. The proof of the second inequality is similar. $\square$

This result implies that if s interpolates a function in $W_p^4(\Omega)$, then the coefficients at level m corresponding to $\lambda_i^m \in \Lambda_k^{(0)}$ will be approximately $1/16$ as large as the analogous coefficients at level $m-1$. Similarly, the coefficients at level m corresponding to $\lambda_i^m \in \Lambda_m^{(1)}$ will be approximately $1/8$ as large as the analogous coefficients at level $m-1$. This observation insures that at higher levels, many coefficients should be small and hence can be removed in the thresholding step.

§7. Numerical Examples

In this section we present some examples to illustrate the performance of the compression scheme. In all cases we choose $\Diamond_0$ as the quadrangulation consisting of the single quadrilateral $Q := [0,1] \times [0,1]$. For each test function f and approximating spline s, we measure both the maximum norm $e_\infty := \|f - s\|$ and the average ℓ_2-norm $e_2 := \|f - s\|_2$.

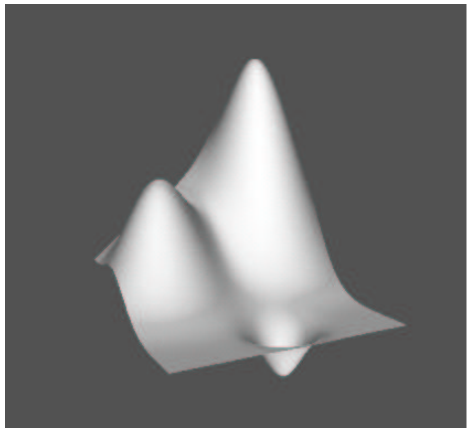
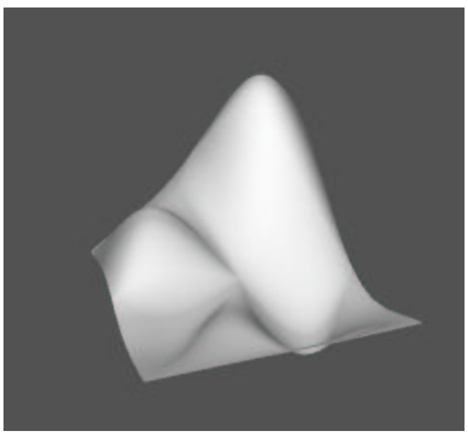


Fig. 2. The function f_1.



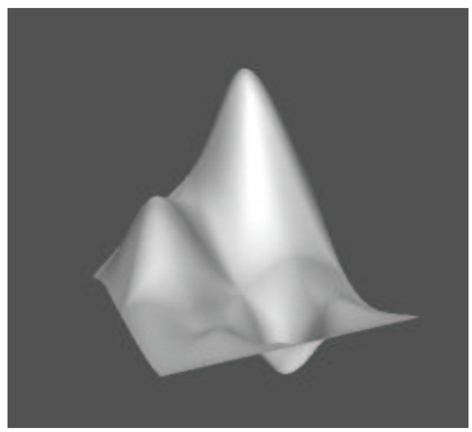


Fig. 3. The splines s_1 and $\hat{s}_6$ fitting f_1.

As a first test function, we take the standard Franke function

$$f_1(x,y) := \frac{3}{4}\left[e^{-\frac{(9x-2)^2}{4}-\frac{(9y-2)^2}{4}} + e^{-\frac{(9x+1)^2}{49}-\frac{(9y+1)}{10}}\right] + \frac{1}{2}e^{-\frac{(9x-7)^2}{4}-\frac{(9y-3)^2}{4}} - \frac{1}{5}e^{-(9x-4)^2-(9y-7)^2}$$

shown in Figure 2. Figure 3 (left) shows the result of interpolating f_1 using a spline s_1 corresponding to level $\ell = 1$. This spline has 39 coefficients and gives errors of $e_\infty = .25$ and $e_2 = .00625$. Figure 3 (right) shows the spline $\hat{s}_6$ which corresponds to interpolating f_1 with a spline s_6 at level 6, and then applying the compression algorithm with $\varepsilon = .02$. Although s_6 has 20,995 coefficients, after compression the spline $\hat{s}_6$ has only 37 coefficients, which corresponds to a compression ratio of 567 to 1. The error bounds for the compressed surface $\hat{s}_6$ are $e_\infty = .099$ and $e_2 = .0008$. Note that although $\hat{s}_6$ has fewer coefficients than s_1, it does a much better job of approximating f_1 and capturing its shape. Both s_1 and $\hat{s}_6$ should be compared

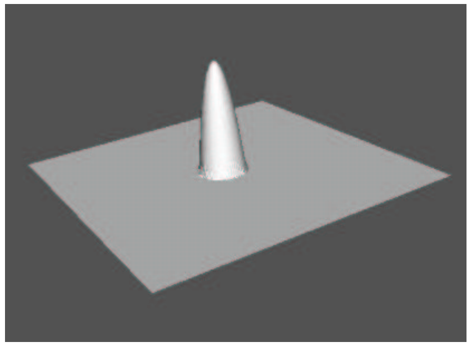


Fig. 4. The surface corresponding to f_2.

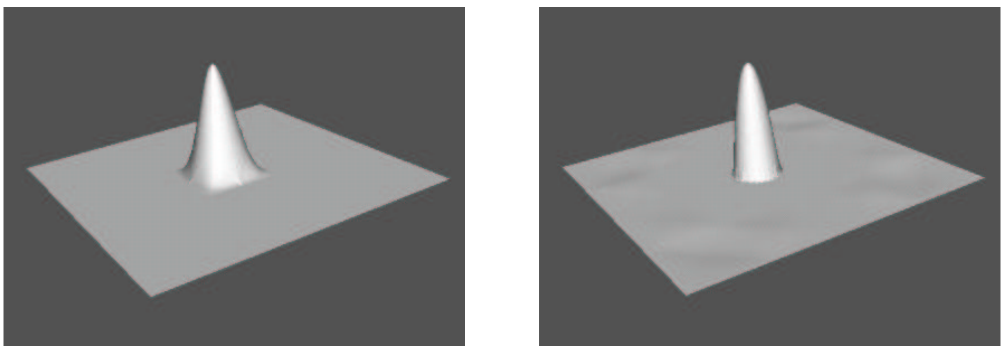


Fig. 5. The splines g_3 and $\hat{g}_6$ fitting f_2.

with the compressed spline approximations of f_1 obtained in [8] which are based on C^0 linear splines. Our surfaces are much smoother since they utilize C^1 cubic splines.

As a second test function we take

$$f_2(x,y) := \begin{cases} e^{-\frac{r_0^2}{r_0^2 - r^2}}, & r < r_0, \\ 0, & \text{otherwise,} \end{cases}$$

where

$$r := r(x,y) = (x - .5)^2 + (y - .5)^2$$

and $r_0 = 1/128$, see Figure 4. Figure 5 (left) shows the result of interpolating f_2 using a spline g_3 corresponding to level $\ell = 3$. This spline has 387 coefficients and gives errors of $e_\infty = .117$ and $e_2 = .000147$. Figure 3 (right) shows the spline $\hat{g}_6$ which corresponds to interpolating f_2 with a spline g_6 at level 6, and then applying the compression algorithm with $\varepsilon = .0023$. The spline g_6 has 20,995 coefficients, but after compression we get $\hat{g}_6$ with only 385 coefficients. The error bounds for the compressed surface $\hat{g}_6$ are $e_\infty = .0074$ and $e_2 = .000004$. Note that g_3 and $\hat{g}_6$ have essentially the same number of coefficients, but $\hat{g}_6$ does a much better job of approximating f_2 and capturing its shape.

§8. Remarks

Remark 8.1. The classical way to create multi-resolution schemes is to work with a nested sequence of spaces $\mathcal{S}_0 \subset \mathcal{S}_1 \subset \cdots \subset \mathcal{S}_\ell$ whose complement spaces $\mathcal{S}_m \ominus \mathcal{S}_{m-1}$ can also be conviently parameterized. Bases for these complement spaces are generally called (pre)-wavelets. While this approach works very well for univariate and tensor-product spline spaces, it becomes quite complicated for bivariate spline spaces built on more general triangulations. Even the case of C^0 linear splines is very complicated, see [8,9] and references therein. Except for box spline spaces (see the following remark), nothing is known for spline spaces with higher order smoothness.

Remark 8.2. Multiresolution schemes have been created for certain box-spline spaces, see e.g. [5,6,16]. Surface compression using C^2 quadratic wavelets was discussed in [5], see also [6].

Remark 8.3. Hierarchical bases are of importance in several areas of mathematics, and in particular in multi-level methods for solving boundary-value problems, see [3,13,14,15,19].

Remark 8.4. The compression ratios reported in Sect. 7 are raw compression ratios. To actually compress a file, we of course have to code the information to show which coefficients have not been thresholded out. This can be done using standard coding techniques. Taking account of this extra overhead leads to lower actual compression ratios.

Remark 8.5. The computation of coefficients of a spline with a hierarchical expansion (5.4) discussed in Theorem 6.1 can be regarded as an example of a Faber interpolation scheme, see [7] and also [2]. Indeed, this expansion corresponds to writing the Hermite interpolating spline $s \in \mathcal{S}_3^1(\diamondplus_\ell)$ in the telescoping form $s = s_0 + (s_1 - s_0) + \cdots + (s - s_{\ell-1})$, where the s_i are given by (6.4). The spline s_i is obtained by interpolating s_{i+1}.

Remark 8.6. We now prove (4.3) for $m = 1$. Let $Q := \langle v_1, v_2, v_3, v_4 \rangle$ be a quadrilateral in $\diamond_0$, and let v_Q be the point where the diagonals of Q intersect. Then Q is divided into four triangles with angles $\alpha_1, \ldots, \alpha_8$ and a_Q, b_Q as shown in Fig. 6. Without loss of generality we may assume $a_Q \le \pi/2 \le b_Q$. Suppose $m_1, \ldots, m_4$ are the midpoints of the sides $\langle v_i, v_{i+1} \rangle$ of Q. After refinement, Q is subdivided into 16 triangles, and as shown in Fig. 7, many of the angles are of exactly the same size as in the original triangulation of Q. In fact the only new angles are $\beta_1, \ldots, \beta_8$. We now show that

$$\beta_i \ge \kappa_Q\, \theta_0, \qquad i = 1, \ldots, 8,$$

where

$$\kappa_Q := \frac{2}{\pi} \sin(a_Q/2).$$

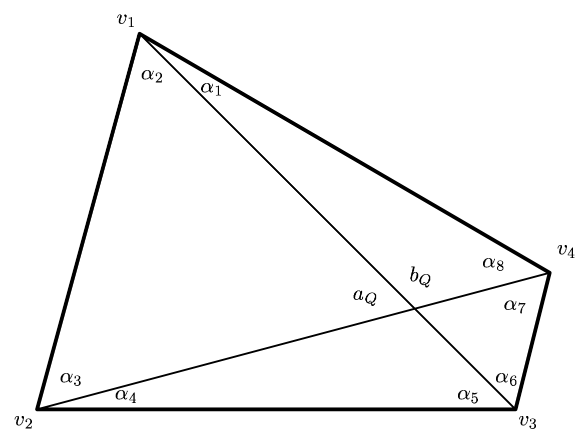


Fig. 6. Angles in a triangulated quadrilateral.

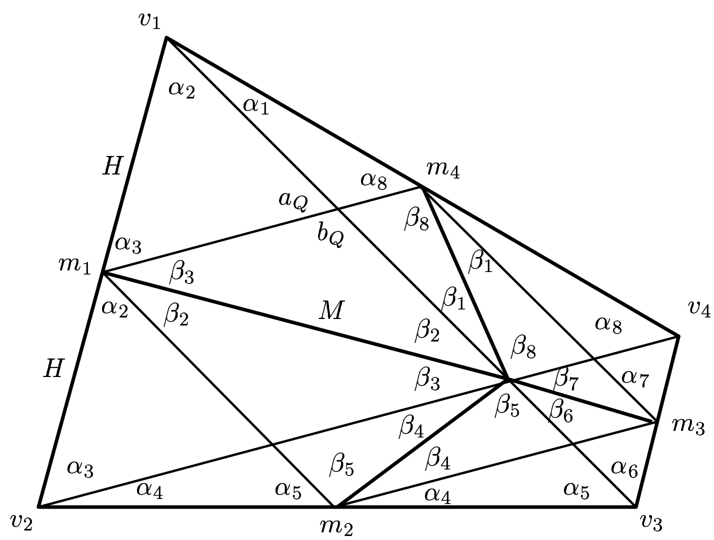


Fig. 7. Angles in a refined quadrilateral.

Since $a_Q \leq \pi/2$ and the line $\langle m_1, m_4 \rangle$ bisects the line $\langle v_1, v_Q \rangle$, it follows that $\tan \beta_1 \geq \tan \alpha_1$ and so $\beta_1 \geq \alpha_1$. A similar argument shows that $\beta_4 \geq \alpha_4$, $\beta_5 \geq \alpha_5$, and $\beta_8 \geq \alpha_8$. We now examine β_2 and β_3 and consider only the case $\beta_2 \leq \beta_3$ as the alternative case is very similar. This implies that $\alpha_2 \leq \pi/2$, and also $\beta_3 \geq a_Q/2$ since $\beta_2 + \beta_3 = a_Q$. Clearly, the edges $\langle v_1, m_1 \rangle$ and $\langle m_1, v_2 \rangle$ have a common length which we denote by H. Let M denote the length of the edge $\langle m_1, v_Q \rangle$. Then by the law of sines,

$$\frac{\sin \beta_2}{H} = \frac{\sin \alpha_2}{M}, \qquad \frac{\sin \beta_3}{H} = \frac{\sin \alpha_3}{M}.$$

Combining these identities with the fact that $\frac{2}{\pi} x \leq \sin x \leq x$ for $0 \leq x \leq \pi/2$, we conclude that

$$\beta_2 \geq \sin\beta_2 = \frac{\sin\beta_3\, \sin\alpha_2}{\sin\alpha_3} \geq \sin(a_Q/2)\, \sin(\theta_0) \geq \frac{2}{\pi}\sin(a_Q/2)\ \theta_0 = \kappa_Q\,\theta_0.$$

But then $\beta_3 \geq \beta_2$ is also greater equal $\kappa_Q\,\theta_0$. A similar argument applies to β_6, β_7. We conclude that the smallest angle in the 16 triangles in the refinement of Q is at least $\kappa_Q\,\theta_0$. Now taking the minimum of κ_Q over all Q in $\diamondsuit_0$, ie.,

$$\kappa := \frac{2}{\pi}\sin(\bar{a}/2), \qquad \bar{a} := \min_{Q\in\diamondsuit_0} a_Q, \tag{8.1}$$

it follows that $\theta_1 \geq \kappa\theta_0$. $\square$

References

1. Ciavaldini, J. F. and J. C. Nedelec, Sur l'élément de Fraeijs de Veubeke et Sander, Rev. Francaise Automat. Informat. Rech. Opér., Anal. Numer. **2** (1974), 29–45.
2. Dæhlen, M., T. Lyche, K. Mørken, R. Schneider, and H.-P. Seidel, Multiresolution analysis over triangles, based on quadratic Hermite interpolation, J. Comput. Appl. Math. **119** (2000), 97–114.
3. Dahmen, W., P. Oswald, and X.-Q. Shi, C^1-hierarchical bases, J. Comput. Appl. Math. **51** (1994), 37–56.
4. Donovan, G. C., J. S. Geronimo, and D. P. Hardin, Compactly supported, piecewise affine scaling functions on triangulations, Constr. Approx. **16** (2000), 201–219.
5. DeVore, R. A., B. Jawerth, and B. Lucier, Surface compression, CAGD **9** (1992), 219–239.
6. DeVore, R. A., B. Jawerth, and V. Popov, Compression of wavelet decompositions, Amer. J. Math. **114** (1992), 737–785.
7. Faber, G., Über stetige Funktionen, Math. Ann. **66** (1909), 81–94.
8. Floater, M. and E. Quak, Piecewise linear prewavelets on arbitrary triangulations, Numer. Math. **82** (1999), 221–252.
9. Floater, M., E. Quak, and M. Reimers, Filter bank algorithms for piecewise linear prewavelets on arbitrary triangulations, J. Comput. Appl. Math. **119** (2000), 185–207.
10. Fraeijs de Veubeke, B., A conforming finite element for plate bending, J. Solids Structures **4** (1968), 95–108.
11. Lai, M. J., Scattered data interpolation and approximation by using bivariate C^1 piecewise cubic polynomials, Comput. Aided Geom. Design **13** (1996), 81–88.

12. Lai, M. J. and L. L. Schumaker, On the approximation power of splines on triangulated quadrangulations, SIAM J. Numer. Anal. **36** (1999), 143–159.

13. Oswald, P., L_p approximation durch Reihen nach dem Haar-Orthogonal-System und dem Faber-Schauder-System, J. Approx. Theory **33** (1981), 1–27.

14. Oswald, P., *Multilevel Finite Element Approximation: Theory and Applications,* Teubner, Stuttgart, 1988.

15. Oswald, P., Hierarchical conforming finite element methods for the biharmonic equation, SIAM J. Numer. Anal. **29** (1992), 1610–1625.

16. Riemenschneider, S. D. and Zuowei Shen, Wavelets and pre-wavelets in low dimensions, J. Approx. Theory **71(1)** (1992), 18–38.

17. Sander, G., Bornes supérieures et inférieures dans l'analyse matricielle des plaques en flexion-torsion, Bull. Soc. Royale Sciences Liège **33** (1964), 456–494.

18. Schröder, P.; and W. Sweldens, Spherical wavelets: efficiently representing functions on the sphere, Comput. Graphics (Proc. Siggraph '95), 161–172.

19. Yserentant, Harry, On the multilevel splitting of finite element spaces, Numer. Math. **49** (1986), 379–412.