Scattered Data Interpolation by Bivariate Splines with Higher Approximation Order*

Tianhe Zhou† and Ming-Jun Lai‡

August 13, 2012

Abstract

Given a set of scattered data, we usually use a minimal energy method to find Lagrange interpolation based on bivariate spline spaces over a triangulation of the scattered data locations. It is known that the approximation order of the minimal energy spline interpolation is only 2 in terms of the size of triangulation. To improve this order of approximation, we propose several new schemes in this paper. Mainly we follow the ideas of clamped cubic interpolatory splines and not-a-knot interpolatory splines in the univariate setting and extend them to the bivariate setting. In addition, instead of the energy functional of the second order, we propose to use higher order versions. We shall present some theoretical analysis as well as many numerical results to demonstrate that our bivariate spline interpolation schemes indeed have a higher order of approximation than the classic minimal energy interpolatory splines.

Keywords and Phrases: bivariate splines, Lagrange spline Interpolation, Hermite spline interpolation, minimal energy method

AMS Classification: 41A15, 65M60, 65N30

---

*This work is supported by Science Foundation of Zhejiang Sci-Tech University (ZSTU) under Grant No. 0813826-Y

†This author is assoiated with Department of Mathematics, Zhejiang Sci-Tech University, 310029 Hangzhou, Zhejiang, China

‡This author is associated with Department of Mathematics, The University of Georgia, Athens, GA 30602, U.S.A. mjlai@math.uga.edu. He is supported by the National Science Foundation under grant #DMS 0713807.

1 Introduction

Suppose $V = \{(x_i, y_i)\}_{i=1}^n$ is a set of data locations lying in a domain $\Omega \subset \mathbf{R}^2$. Let $\triangle$ be a triangulation of the data locations. Let $\{z_i, i = 1, \cdots, n\}$ be given real values. We would like to construct a smooth function $s \in C^r(\Omega)$ with $r \geq 1$.

$$s(v_i) = z_i, i = 1, \ldots, n. \tag{1.1}$$

We shall use the space of polynomial splines

$$S_d^r(\triangle) := \{s \in C^r(\Omega) : s|_T \in \mathcal{P}_d, \quad \forall T \in \triangle\},$$

where $d > r$ is a given integer, $\mathcal{P}_d$ is the space of bivariate polynomials of degree d, and $\triangle$ is a triangulation of the given data locations. This problem is known as bivariate spline Lagrange interpolation. A classic solution to this problem is the so-called minimal energy method (cf., e.g. [3]) which finds the spline $s^* \in S_5^1(\triangle)$ satisfying (1.1) such that

$$E_2(s^*) = \min\{E_2(s) : s(v_i) = z_i, i = 1, \ldots, n, s \in S_5^1(\triangle)\},$$

where

$$E_2(s) = \int_\Omega [s_{xx}^2 + 2s_{xy}^2 + s_{yy}^2] dxdy \tag{1.2}$$

is called thin-plate energy functional. Of course, we can find other methods to do Lagrange interpolation without minimizing an energy functional (cf., e.g. [13],[14],[15]). For example, in [14], the researchers use C^1 splines based on Clough-Tocher triangulation to do interpolation and also point out that their Lagrange interpolation schemes possess the optimal approximation order. Indeed, they combined extra smoothness conditions, Clouch-Tocher splitting technique, and interpolation conditions to locally determine the MDS(minimal determine set) around each triangle and then fix all the remaining coefficients of a bivariate interpolatory spline. As we use the minimal energy method to globally fix all the extra coefficients besides interpolatory conditions, the surface of interpolatory spline created by the minimal energy method has minimal variation and oscillations. It is known that the approximation order of the interpolatory spline obtained by the minimal energy method is 2 in terms of the size of triangulation (cf. [4]). In [4], the researchers explained that the order of approximation will not increase even if one increases the degree of spline spaces. A numerical experiment is provided to show that the order is only 2 for different degrees.

How to increase the approximation order when doing scattered data interpolation is the main motivation of this paper. One approach is to interpolate derivative values in addition to function values. In [12], bivariate Hermite interpolatory splines were studied. The authors of their paper [12] established the approximation order of the bivariate spline Hermite interpolation scheme. The approximation order is indeed increased. See Theorem 2.2 for a special case $m = 3$. More precisely, for any integer $m \geq 2$, let

$$E_m(f) = \int_\Omega \left[\sum_{k=0}^{m} \binom{m}{k} \left[(D_x)^k (D_y)^{m-k} f \right]^2 \right] dxdy \tag{1.3}$$

be a general energy functional. A Hermite interpolatory spline $s^* \in S_d^{m-1}(\triangle)$ for an appropriate d, e.g. $d \geq 3m - 1$ satisfying

$$D_x^\alpha D_y^\beta s^*(x_i, y_i) = f_{i,\alpha,\beta}, \quad \alpha + \beta \leq m - 2, \tag{1.4}$$

such that

$$E_m(s^*) = \min\{E_m(s), \quad s \in S_d^{m-1}(\triangle), s \text{ satisfies } (1.4)\}.$$

Although such Hermite interpolatory splines have a higher order of approximation, in practice, we may not have these derivatives at all vertices or it needs a lot of effort and/or high cost to collect these derivative values.

The purpose of this paper is to construct several interpolatory spline schemes without using all derivative information. Recall as observed in the end of the paper [4], the error behavior is similar to the well-known natural cubic spline which minimizes the univariate energy $\int_a^b [s''(x)]^2 dx$ among all smooth functions that interpolate given values at points $a = x_0 < \cdots < x_n = b$. That is, numerical experiments show that the approximation of a minimal energy spline is better inside the underlying domain than near the boundary. It is also well-known that the full cubic spline space (with no special boundary conditions) has approximation power $\mathcal{O}(h^4)$ where h is the mesh size, but the interpolating natural spline only has approximation order $\mathcal{O}(h^2)$. This loss of accuracy is due to the natural boundary conditions, and indeed the interpolating spline does exhibit $\mathcal{O}(h^4)$ accuracy in a compact subset of $[a, b]$ which stays away from the boundary. Carl de Boor suggested that the analogous situation might also hold for bivariate minimal energy splines (cf. [4]). But the bivariate spline space is much more complicated than univariate spline space. There are too many extra coefficients besides the interpolatory conditions need to be fixed in the bivariate spline setting. So we have to use the minimal energy method or other methods to solve this situation. Thus we propose the following new bivariate interpolatory spline schemes.

Clamped Interpolation Scheme: We find the spline function $s^* \in S_d^2(\triangle)$ satisfying the interpolation conditions (1.1) as well as boundary Hermite interpolation conditions

$$D_x^\alpha D_y^\beta s^*(x_i, y_i) = f_{i,\alpha,\beta}, \quad \alpha + \beta \leq m - 2, (x_i, y_i) \in \partial\Omega, \tag{1.5}$$

which minimizes E_3, where $\partial\Omega$ denotes the boundary of Ω. Here $d \geq 8$ if $\triangle$ is a general triangulation or appropriate d if $\triangle$ is a Cough-Tocher or Powell-Sabin refinement of triangulation or a FVS triangulation. See, e.g., [11].

Numerical experiments and a theoretical study show that the approximation order of this scheme is comparable to that of the bivariate Hermite spline interpolation discussed in [12], where the Hermite interpolatory splines use the derivatives at all vertices. Thus our clamped interpolatory splines are better in the sense that we use only derivatives at boundary vertices.

Again we face the challenge that we may not have boundary derivatives available for general real-life practical problems. Thus, we propose three different approaches to overcome this difficulty in Sect. 4.

Lagrange Interpolation Scheme with E_3: The easiest approach is do nothing. That is, we find an $s^* \in S_8^2(\triangle)$ satisfying (1.1) which minimizes higher order energy functional E_3. Our numerical experiments clearly show that Lagrange interpolation using E_3 is much better than that of using E_2.

Least Squares Scheme: We construct a least squares polynomial fitting to function values nearby a boundary vertex v_b and use its derivative to approximate the true derivatives at v_b. Then we use clamped spline interpolation discussed avove. We will explain this approach in more details in Sect. 4.2.

Multiple Point Scheme: In this approach, we propose to use the multiple point method to estimate the derivative value at all boundary vertices. Once having these estimating boundary derivative values, we can clamp down the spline interpolation. Again we will discuss this scheme in Sect. 4.3.

The above two schemes are motivated by the Clamped C^2 cubic interpolatory splines. Numerical examples show that it has a higher approximation order. We should also explore the ideas of not-a-knot splines in the univariate setting (cf. [2]). This leads to

Boundary CT Scheme: Given a triangulation $\triangle$ of all data locations, we first refine all the triangles in $\triangle$ by Clough-Tocher refinement to get a new triangulation $\triangle_{CT}$. The boundary CT scheme is to find an interpolatory spline on $\triangle_{CT}$ by adding smoother conditions on the boundary triangles to force the reproduction of cubic polynomials on boundary triangles in $\triangle$. We will explain more detail in Sect. 5. Numerical examples show that this scheme is better than Lagrange interpolation.

The paper is organized as follows. In Sect. 2 we briefly review some well-known Bernstein-Bézier notation. More details can be found in [11]. In Sect. 3 we present the Clamped interpolation scheme together with numerical examples to demonstrate their advantage. Also we give some theoretical study to justify the higher order approximation is possible in the same section. In Sect. 4 we discuss several approaches of estimating the derivatives using function values. Then we present some numerical examples to show the advantage of each scheme in the same Section. In Sect. 5, we shall explain the boundary CT schemes. Finally, we give some remarks about the approximation property of interpolation schemes in Sect. 6.

2 Preliminaries

Given a triangulation $\triangle$ and integers $0 \le r < d$, we write

$$S_d^r(\triangle) := \{s \in C^r(\Omega) : s|_T \in P_d, \text{for all } T \in \triangle\}$$

for the usual space of splines of degree d and smoothness r, where P_d is the $\binom{d+2}{2}$ dimensional space of bivariate polynomials of degree d. Throughout the paper we shall make extensive use of the well-known Bernstein-Bézier representation of splines. For each triangle $T = \langle v_1, v_2, v_3 \rangle$ in $\triangle$ with vertices v_1, v_2, v_3, the corresponding polynomial piece $s|_T$ is written in the form

$$s|_T = \sum_{i+j+k=d} c_{ijk}^T B_{ijk}^d,$$

where B_{ijk}^d are the Bernstein-Bézier polynomials of degree d associated with T. In particular, if $(\lambda_1, \lambda_2, \lambda_3)$ are the barycentric coordinates of any point $u \in \mathbf{R}^2$ in term of the triangle T, then

$$B_{ijk}^d(u) := \frac{d!}{i!j!k!} \lambda_1^i \lambda_2^j \lambda_3^k, \quad i + j + k = d$$

Usually, we associate the *Bernstein-Bézier coefficients* $\{c_{ijk}^T\}_{i+j+k=d}$ with the domain points $\{\xi_{ijk}^T := (iv_1 + jv_2 + kv_3)/d\}_{i+j+k=d}$.

Definition 2.1 *Let $\beta < \infty$. A triangulation $\triangle$ is said to be β-quasi-uniform provided that $|\triangle| \le \beta \rho_\triangle$, where $|\triangle|$ is the maximum of the diameters of the triangles in $\triangle$, and $\rho_\triangle$ is the minimum of the radii of the incircles of triangles of $\triangle$.*

It is easy to see that if $\triangle$ is β-quasi-uniform, then the smallest angle in $\triangle$ is bounded below by $2/\beta$.

Recall that a *determining set* for a spline space $S \subseteq S_d^0(\triangle)$ is a subset $\mathcal{M}$ of the set of domain points such that if $s \in S$ and $c_\xi = 0$ for all $\xi \in \mathcal{M}$, then $c_\xi = 0$ for all domain points. The set $\mathcal{M}$ is called a *minimal determining set (MDS)* for S if there is no smaller determining set. It is known that $\mathcal{M}$ is a

MDS for S if and only if every spline $s \in S$ is uniquely determined by its set of B-coefficients$\{c_\xi\}_{\xi \in \mathcal{M}}$.

Suppose that $T := \langle v_1, v_2, v_3 \rangle$ and $\widehat{T} := \langle v_4, v_3, v_2 \rangle$ are two adjoining triangles from $\triangle$ which share the edge $e := \langle v_2, v_3 \rangle$, and let

$$s|_T = \sum_{i+j+k=d} c_{ijk} B^d_{ijk},$$

$$s|_{\widehat{T}} = \sum_{i+j+k=d} \widehat{c}_{ijk} \widehat{B}^d_{ijk},$$

where B^d_{ijk} and $\widehat{B}^d_{ijk}$ are the Bernstein polynomials of degree d on the triangles T and $\widehat{T}$, respectively. Given integers $0 \leq n \leq j \leq d$, let $\tau^n_{j,e}$ be the linear functional defined on $S^0_d(\triangle)$ by

$$\tau^n_{j,e} s := c_{n,d-j,j-n} - \sum_{\nu+\mu+\kappa=n} \widehat{c}_{\nu,\mu+j-n,\kappa+d-j} \widehat{B}^n_{\nu\mu\kappa}(v_1).$$

It is called *smoothness functional of order* n. Clearly a spline $s \in S^0_d(\triangle)$ belongs to $C^r(\Omega)$ for some $r > 0$ if and only if

$$\tau^n_{m,e} s = 0, \quad n \leq m \leq d, \quad 0 \leq n \leq r$$

So we shall often make use of smoothness conditions to calculate one coefficient of a spline in terms of others.

About the approximation order of interpolatory spline spaces, the following result can be found in [12].

Theorem 2.2 *Suppose $\triangle$ is a β-quasi-uniform triangulation and $f \in C^3(\Omega)$. Then there exists a constant K depending only on d, β and f such that the Hermite interpolant $s_f \in S^2_d(\triangle)$ satisfying (1.4) with $m = 3$ possesses*

$$\|f - s_f\|_{L_\infty(\Omega)} \leq K |\triangle|^3 |f|_{3,\infty,\Omega},$$

where $|\triangle|$ is the mesh size of $\triangle$ (ie., the diameter of the largest triangle), and $|f|_{3,\infty,\Omega}$ is the maximum norm of the 3th derivatives of f over Ω.

When $d < 8$, similar approximation results are available for some special spline spaces, see [5], [7], [8], [9] and [10].

3 Clamped Interpolation Scheme

Recall that an energy functional $E_m(f)$ is an expression for the amount of potential energy in a thin elastic plate f that passes through the data points V. In this scheme, we use the energy functional $E_3(f)$ in (1.3) when $m = 3$. That is

$$E_3(f) = \int_\Omega \left[\sum_{k=0}^{3} \binom{3}{k} \left[(D_x)^k (D_y)^{3-k} f \right]^2 \right] dxdy. \tag{3.1}$$

Let V_B be the list of all boundary vertices. Thus the clamped interpolation scheme can be formulated as follows: find a spline $s^* \in S_d^2(\triangle)$ such that

$$s^*(v_i) = z_i, D_x s^*(v_j) = z_j^{1,0}, D_y s^*(v_j) = z_j^{0,1}, v_i \in V, v_j \in V_B \tag{3.2}$$

and

$$\begin{aligned} E_3(s^*) \;\; = \;\; & \min\{E_3(s) : s(v_i) = z_i, D_x s(v_j) = z_j^{1,0}, \\ & D_y s(v_j) = z_j^{0,1}, v_i \in V, v_j \in V_B, s \in S_d^2(\triangle)\}. \end{aligned} \tag{3.3}$$

Note that the above scheme can be implemented using the approach proposed in [1]. Before we present a theoretical study the approximation power of this scheme, we first demonstrate several numerical experiments by comparing the approximation orders of minimal energy splines and Hermite interpolatory splines for different test functions.

Example 3.1 (a) *Suppose $\diamondsuit$ is a uniform partition of the unit square domain $\Omega := [0,1] \times [0,1]$ into N^2 subsquares. Let $S_8^2(\triangle_\diamondsuit)$ be a C^2 spline space, where $\triangle_\diamondsuit$ is the triangulation obtained by inserting one diagonal of each subsquare in $\diamondsuit$. We use the following test functions:*

$$\begin{aligned} f_1(x,y) &= (x+1)^3 + (y+1)^3 \\ f_2(x,y) &= \sin(2(x-y)) \\ f_3(x,y) &= 0.75\exp(-0.25(9x-2)^2 - 0.25(9y-2)^2) \\ &+ 0.75\exp(-(9x+1)^2/49 - (9y+1)/10) \\ &+ 0.5\exp(-0.25(9x-7)^2 - 0.25(9y-3)^2) \\ &- 0.2\exp(-(9x-4)^2 - (9y-7)^2), \end{aligned}$$

where f_3 is the well-known Franke function. We use these function values and derivative values at the grid points $(i/N, j/N), i, j = 0, \cdots, N$ to have a set of scattered Hermite data. We approximated these functions for choices $N = 4, 8, 16, 32$ which corresponds to repeatedly halving the mash size (see Fig 1) using three different methods: classic minimal energy method for Lagrange interpolation using $S_5^1(\triangle)$ (cf. [1]), minimal energy method for Hermite Interpolation using $S_8^2(\triangle)$ (cf. [12]) and our proposed clamped spline interpolation in $S_8^2(\triangle)$. Table 1(a). gives the maximum errors for all these three methods computed based on 201×201 equally-spaced points over Ω.

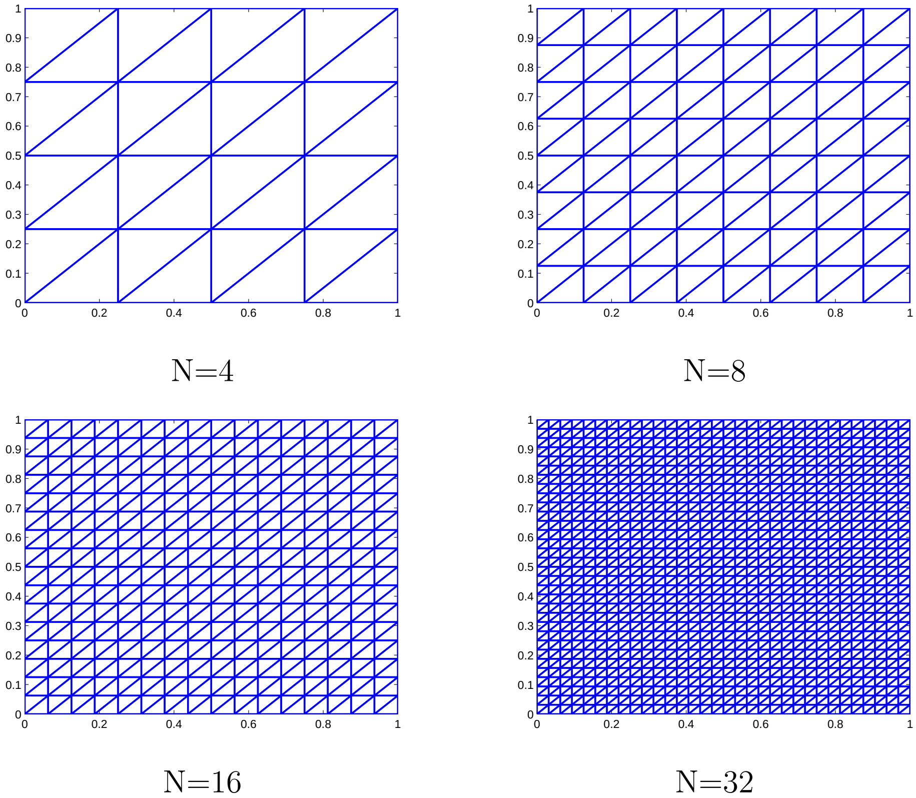


Figure 1: Triangulations (a) for choice $N = 4, 8, 16, 32$

$f_1 \ \backslash \ N$	4	8	16	32
Lagrange Interp.	$6.85e-002$	$1.70e-002$	$4.20e-003$	$1.04e-003$
Hermite Interp.	$2.55e-004$	$3.18e-005$	$3.95e-006$	$3.78e-007$
Clamped Interp.	$5.95e-004$	$6.12e-005$	$7.56e-006$	$9.80e-007$
$f_2 \ \backslash \ N$	4	8	16	32
Lagrange Interp.	$2.38e-002$	$5.32e-003$	$1.31e-003$	$3.20e-004$
Hermite Interp.	$2.74e-004$	$3.15e-005$	$3.69e-006$	$3.19e-007$
Clamped Interp.	$4.27e-004$	$7.37e-005$	$9.92e-006$	$6.58e-007$
$f_3 \ \backslash \ N$	4	8	16	32
Lagrange Interp.	$9.28e-002$	$5.01e-002$	$4.12e-003$	$5.63e-004$
Hermite Interp.	$4.15e-002$	$7.64e-003$	$2.05e-004$	$6.68e-005$
Clamped Interp.	$8.13e-002$	$3.62e-002$	$7.44e-004$	$8.24e-005$

Table 1(a). Maximum errors for various test functions

(b) *Although the above numerical results are based on uniform triangulations, similar numerical results can be obtained for arbitrary triangulations.*

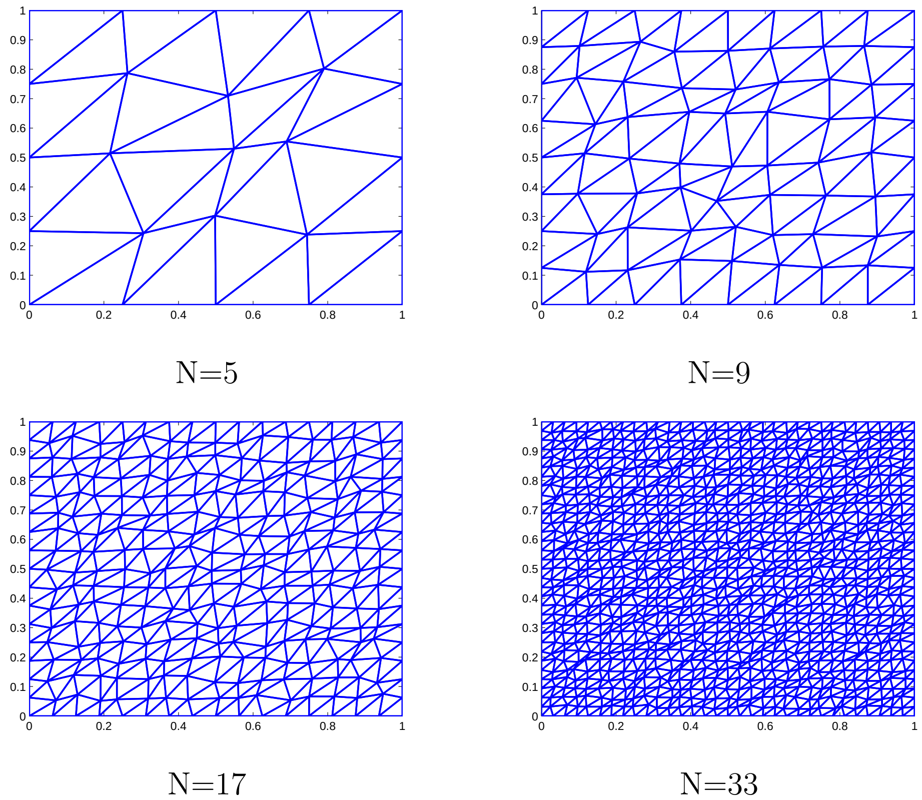


Figure 2: Triangulations (b) for choice $N = 5, 9, 17, 33$

For example, we perturb the triangulations in Fig. 1 to have new triangulations as shown in Fig.2.

$f_1 \setminus N$	5	9	17	33
Lagrange Interp.	$6.16e-002$	$2.26e-002$	$4.04e-003$	$1.16e-003$
Hermite Interp.	$3.85e-004$	$6.43e-005$	$6.72e-006$	$1.97e-006$
Clamped Interp.	$8.16e-004$	$1.02e-004$	$1.14e-005$	$5.66e-006$
$f_2 \setminus N$	5	9	17	33
Lagrange Interp.	$2.22e-002$	$6.61e-003$	$1.44e-003$	$3.06e-004$
Hermite Interp.	$2.72e-004$	$5.65e-005$	$2.14e-005$	$6.48e-006$
Clamped Interp.	$5.29e-004$	$1.32e-004$	$4.67e-005$	$9.01e-006$
$f_3 \setminus N$	5	9	17	33
Lagrange Interp.	$1.19e-001$	$8.54e-002$	$4.98e-003$	$8.43e-004$
Hermite Interp.	$4.11e-002$	$2.26e-002$	$3.14e-004$	$8.15e-005$
Clamped Interp.	$1.21e-001$	$6.19e-002$	$7.51e-004$	$1.77e-004$

Table 1(b). Maximum errors for various test functions

We use the same testing functions and same three methods to show that the approximation order of these splines are not dependent whether they are uniform or not. Table 1(b) gives the maximum errors of these three spline interpolatory schemes computed based on 201×201 *equally-spaced points over domain.*

Discussion: *From Tables 1(a) and 1(b) we can see the maximum error of clamped interpolation is almost the same as that of Hermite interpolation. This show that our clamped interpolatory splines has an advantage over the Hermite interpolatory splines. Thus we recommend to use boundary Hermite interpolation(clamped interpolation) instead of the global Hermite interpolation when functions to be interpolated are smooth functions.*

We now study the approximation order of our clamped interpolatory splines. First we convert the clamped spline interpolation problem (3.3) into a standard approximation problem in Hilbert space. Let

$$X := \{f \in B(\Omega) : f|_T \in W^2_\infty(T), \text{all triangles } T \text{ in } \triangle\}$$

where $B(\Omega)$ is the set of all bounded real-valued functions on Ω. For each triangle T in $\triangle$, Let

$$\langle f, g\rangle_{X_T} := \int_T \left[\sum_{k=0}^{3} \binom{3}{k} \left[(\frac{\partial}{\partial x})^k (\frac{\partial}{\partial y})^{3-k} f (\frac{\partial}{\partial x})^k (\frac{\partial}{\partial y})^{3-k} g\right]\right] dxdy.$$

Then the following

$$\langle f, g\rangle_X := \sum_{T\in\triangle} \langle f, g\rangle_{X_T}$$

defines a semi-definite inner-product on X. Let $\|f\|_{X_T}$ and $\|f\|_X$ be the associated semi-norms.

Suppose $\mathcal{S} \subseteq S^2_d(\triangle)$ is a spline subspace on a triangulation $\triangle$, and that $\mathcal{S}$ has a stable local basis $\{B_\xi\}_{\xi\in\mathcal{M}}$ corresponding to a minimal determining set $\mathcal{M}$ containing the set of vertices V of $\triangle$ as explained in [11]. Let V_B be the collection of the boundary vertices, then it is easy to see the $\langle\cdot,\cdot\rangle_X$ is an inner-product on the linear space

$$W := \{s \in \mathcal{S} : s(v) = 0, v \in V, \frac{\partial}{\partial x} s(v_b) = 0, \frac{\partial}{\partial y} s(v_b) = 0, v \in V, v_b \in V_B\}. \tag{3.4}$$

Given a triangle T, let $\text{star}^0(T) = T$, and

$$\text{star}^q(T) := \bigcup\{T \in \triangle : T \cap \text{star}^{q-1}(T) \neq \emptyset\}, \qquad q \geq 1.$$

Lemma 3.2 *Suppose* $\langle w, w\rangle_X = 0$ *for some* $w \in W$*, Then* $w = 0$.

Proof: Since $\langle w, w\rangle_X = 0$ for some $w \in W$, then w is function of degree 2 on $\triangle$. Let $T_B =< v_1, v_2, v_3 >$ be a boundary triangle with boundary vertices $v_1, v_2 \in V_B$. First we use $w(v_i) = 0, i = 1, 2, 3$ to determine the domain points at the vertices. Next we can determine the last three domains points ξ_{110}, ξ_{101} and ξ_{011} by $w_x(v_i) = 0$ and $w_y(v_i) = 0$ for $i = 1, 2$. Then for any triangle $T \in \text{star}(T_B)$, we can use the C^1 smoothness condition to determine the domain points ξ_{110}, ξ_{101} and ξ_{011} in T. Remember that remaining domain points can be determined by function w vanishes at all vertices. Using the this way, we can see $w = 0$ on all boundary triangles. The C^1 conditions imply that w at the vertices v connected to boundary triangles satisfies

$$\frac{\partial}{\partial x} w(v) = 0, \quad \frac{\partial}{\partial y} w(v) = 0$$

in addition to $w(v) = 0$. We can repeat the above arguments to see $w = 0$ over triangles next to the boundary triangles. In this way, we see $w = 0$ on in any triangle in triangulation $\triangle$. So it follows that $w \equiv 0$. ♠

According to the Lemma 3.2, then W equipped with the inner-product $\langle \cdot, \cdot \rangle_X$ is a Hilbert space. Let

$$\begin{aligned} U_f := \{s \in \mathcal{S} : s(v) &= f(v), v \in V, \frac{\partial}{\partial x} s(v_b) = f_x(v_b), \\ \frac{\partial}{\partial x} s(v_b) &= f_y(v_b), v \in V, v_b \in V_B\} \end{aligned} \tag{3.5}$$

be set of all splines in $\mathcal{S}$ that interpolate function value at the points of V and derivative value at points of V_B. Then we choose a spline $S_f \in \mathcal{S}$ such that

$$E_3(S_f) = \min_{s \in U_f} E_3(s) \tag{3.6}$$

Given f, suppose s_f is any spline in the set U_f defined in (3.5). Then it is easy to see that the solution $S_f \in \mathcal{S}$ to the clamped spline interpolation problem (3.6) is equal to $s_f - Ps_f$, where P is the linear projector $P : X \to W$ defined by

$$E_3(g - Pg) = \min_{w \in W} E_3(g - w)$$

for all $g \in X$. Since W is a Hilbert space with respect to $\langle \cdot, \cdot \rangle_X$, Pg is uniquely defined, and is characterized by

$$\langle g - Pg, w \rangle_X = 0, \quad \text{for all } w \in W.$$

Moreover, using the Cauchy-Schwarz inequality, it is easy to see that

$$\|Pg\|_X \le \|g\|_X$$

for all $g \in X$.

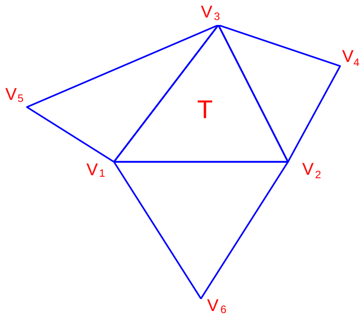


Figure 3: F(T): Union of T and its three neighboring triangles

Lemma 3.3 *Let $T = \langle v_1, v_2, v_3 \rangle$ be an interior triangle in a triangulation $\triangle$ in the sense that there exist three triangles in $\triangle$ each of which shares a common edge with T. Let $F(T)$ be the union of T and the three neighboring triangles. Let v_{i+3} be the opposite vertex of v_i in $F(T)$ for $i = 1, 2, 3$. See Fig. 3. Suppose that all six vertices $v_i, i = 1, \cdots, 6$ do not lie on the locus of any bivariate quadratic polynomial. Suppose that $f \in W^2_\infty(T)$ satisfies*

$$f(v_i) = 0$$

for $i = 1, \ldots, 6$. Then for all $v \in T$,

$$|f(v)| \leq C_1 |T|^3 |f|_{3,\infty,F(T)}. \tag{3.7}$$

Proof: The assumed hypotheses on the six vertices immediately imply the existence and uniqueness of bivariate quadratic polynomial verifying the conditions $p(v_i) = g_i, i = 1, 2, \cdots, 6$.

Let $K := \dfrac{|F(T)|}{|T|}$ be the ratio of the length of edge in star(T) and in T. Given $v \in F(T)$, we can write $v = v_1 + t(v_2 - v_1) + u(v_3 - v_1)$ with $|t| \leq K$ and $|u| \leq K$. Let $g(t, u) = f(v_1 + t(v_2 - v_1) + u(v_3 - v_1))$ for $v \in \text{star}(T)$. It is easy to see

$$\begin{array}{lll} f(v_1) = g(0,0), & f(v_2) = g(1,0), & f(v_3) = g(0,1) \\ f(v_4) = g(t_1, u_2), & f(v_5) = g(t_2, u_2), & f(v_6) = g(t_3, u_3) \end{array}$$

for some fixed constants t_i and u_i for $i = 1, 2, 3$. Since $f(v_i) = 0$ for $i = 1, \ldots, 6$, we can get $g(0,0) = 0$, $g(1,0) = 0$, $g(0,1) = 0$, $g(t_1, u_1) = 0$, $g(t_2, u_2) = 0$, $g(t_3, u_3) = 0$. By Taylor's expansion, we have

$$0 = g(1,0) = g(0,0) + g_t(0,0) + \frac{1}{2} g_{tt}(0,0) + R_1, \tag{3.8}$$

$$0 = g(0,1) = g(0,0) + g_u(0,0) + \frac{1}{2} g_{uu}(0,0) + R_2, \tag{3.9}$$

$$\begin{aligned} 0 = g(t_1, u_1) &= g(0,0) + t_1 g_t(0,0) + u_1 g_u(0,0) + \frac{1}{2} t_1^2 g_{tt}(0,0) \\ &+ t_1 u_1 g_{tu}(0,0) + \frac{1}{2} u_1^2 g_{uu}(0,0) + R_3, \end{aligned} \tag{3.10}$$

$$\begin{aligned} 0 = g(t_2, u_2) &= g(0,0) + t_2 g_t(0,0) + u_2 g_u(0,0) + \frac{1}{2} t_2^2 g_{tt}(0,0) \\ &+ t_2 u_2 g_{tu}(0,0) + \frac{1}{2} u_2^2 g_{uu}(0,0) + R_4, \end{aligned} \tag{3.11}$$

and

$$\begin{aligned} 0 = g(t_3, u_4) &= g(0,0) + t_3 g_t(0,0) + u_3 g_u(0,0) + \frac{1}{2} t_3^2 g_{tt}(0,0) \\ &+ t_3 u_3 g_{tu}(0,0) + \frac{1}{2} u_3^2 g_{uu}(0,0) + R_5, \end{aligned} \tag{3.12}$$

where $R_1, \ldots, R_5$ are remainder terms. It easy to see $R_i = K_i |g|_{3,\infty,\mathrm{star}(T)}$ for some constants $K_i, i = 1, \ldots, 5$. We have

$$\begin{pmatrix} 1 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 & 0 & \frac{1}{2} \\ t_1 & u_1 & \frac{1}{2}t_1^2 & t_1u_1 & \frac{1}{2}u_1^2 \\ t_2 & u_2 & \frac{1}{2}t_2^2 & t_2u_2 & \frac{1}{2}u_2^2 \\ t_3 & u_3 & \frac{1}{2}t_3^2 & t_3u_3 & \frac{1}{2}u_3^2 \end{pmatrix} \begin{pmatrix} g_t(0,0) \\ g_u(0,0) \\ g_{tt}(0,0) \\ g_{tu}(0,0) \\ g_{uu}(0,0) \end{pmatrix} = |g|_{3,\infty,\mathrm{star}(T)} \begin{pmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \end{pmatrix}$$

It is easy to see, the matrix above is invertible since the assumption. We can get

$$\begin{aligned} |g_t(0,0)| &\le K_6 |g|_{3,\infty,\mathrm{star}(T)}, \\ |g_u(0,0)| &\le K_7 |g|_{3,\infty,\mathrm{star}(T)}, \\ |g_{tt}(0,0)| &\le K_8 |g|_{3,\infty,\mathrm{star}(T)}, \\ |g_{tu}(0,0)| &\le K_9 |g|_{3,\infty,\mathrm{star}(T)}, \\ |g_{uu}(0,0)| &\le K_{10} |g|_{3,\infty,\mathrm{star}(T)} \end{aligned}$$

where $K_i, i = 6, \ldots, 10$ are positive constants. Thus

$$\begin{aligned} |f(v)| &= |g(t,u)| \le |g(0,0)| + K|g_t(0,0)| + K|g_u(0,0)| \\ &+ \frac{1}{2}K^2 |g_{tt}(0,0)| + K^2 |g_{tu}(0,0)| + \frac{1}{2} K^2 |g_{uu}(0,0)| + K_{11} |g|_{3,\infty,\mathrm{star}(T)} \\ &\le K_{12} |g|_{3,\infty,\mathrm{star}(T)}. \end{aligned}$$

Since $|g|_{3,\infty,\mathrm{star}(T)} \le K_{13} |f|_{3,\infty,\mathrm{star}(T)} |\mathrm{star}(T)|^3 = K^3 K_{13} |f|_{3,\infty,\mathrm{star}(T)} |T|^3$, we conclude that (3.7) holds. ♠

Lemma 3.4 *Let $T = \langle v_1, v_2, v_3 \rangle$ be a boundary triangle with boundary vertex v_1 in triangulation $\triangle$ and v_4 be the opposite vertex of v_1 in neighbor triangle of T. Suppose that $f \in W^2_\infty(T)$ satisfies*

$$f(v_i) = 0, i = 1, 2, 3, 4 \text{ and } f_x(v_1) = 0 \quad f_y(v_1) = 0.$$

Then for all $v \in T$,

$$|f(v)| \le C_2 |T|^3 |f|_{3,\infty,\text{star}(T)}. \tag{3.13}$$

Proof: Let $K := \dfrac{|F(T)|}{|T|}$ be the ratio of the length of edge in star(T) and in T. Given $v \in \text{star}(T)$, we can write $v = v_1 + t(v_2 - v_1) + u(v_3 - v_1)$ with $|t| \le K$ and $|u| \le K$. Let $g(t, u) = f(v_1 + t(v_2 - v_1) + u(v_3 - v_1))$ for $v \in \text{star}(T)$. It is easy to see

$$\begin{array}{ll} f(v_1) = g(0,0), & f(v_2) = g(1,0), \\ f(v_3) = g(0,1) & f(v_4) = g(t_1, u_2), \end{array}$$

for some fixed constants t_i and u_i for $i = 1, \ldots, 4$. Since $f_x(v_1) = 0$, $f_y(v_1) = 0$ and $f(v_i) = 0$ for $i = 1, \ldots, 4$, we can get $g(0,0) = 0$, $g(1,0) = 0$, $g(0,1) = 0$, $g(t_1, u_1) = 0$, $g_t(0,0) = 0$, $g_u(0,0) = 0$. By Taylor's expansion, we have

$$0 = g(1,0) = g(0,0) + g_t(0,0) + \frac{1}{2} g_{tt}(0,0) + R_1, \tag{3.14}$$

$$0 = g(0,1) = g(0,0) + g_u(0,0) + \frac{1}{2} g_{uu}(0,0) + R_2, \tag{3.15}$$

and

$$\begin{aligned} 0 = g(t_1, u_1) &= g(0,0) + t_1 g_t(0,0) + u_1 g_u(0,0) + \frac{1}{2} t_1^2 g_{tt}(0,0) \\ &+ t_1 u_1 g_{tu}(0,0) + \frac{1}{2} u_1^2 g_{uu}(0,0) + R_3, \end{aligned} \tag{3.16}$$

where $R_1, \ldots, R_3$ are remainder terms. Next we can use the same method in Lemma 3.3 to get (3.13). ♠

Recall from [12], we have

Theorem 3.5 *Suppose $\triangle$ is a β-quasi-uniform triangulation $\triangle$. Suppose that $\{B_\xi\}_{\xi \in \mathcal{M}}$ is a stable local basis for $S^2_d(\triangle)$ with $d \ge 8$ corresponding to a minimal determining set $\mathcal{M}$ containing the set V of vertices of $\triangle$. Then*

$$|Pg|_{3,\infty,\Omega} \le C_3 |g|_{3,\infty,\Omega} \qquad \text{for all } g \in X, \tag{3.17}$$

where C_3 depends only on d,l, r, and β.

We are now ready to prove the main theoretical result in this paper:

Theorem 3.6 *Suppose $\triangle$ is a β-quasi-uniform triangulation. Suppose that $f \in C^3(\Omega)$. Then there exists a constant C depending only on d, β and f such that the clamped spline interpolant S_f defined in (3.6) satisfies*

$$\|f - S_f\|_{L_\infty(\Omega)} \le C|\triangle|^3 |f|_{3,\infty,\Omega} \tag{3.18}$$

Proof: Given a function $f \in C^3(\Omega)$, let $s_f \in U_f$ be the Hermite interpolant spline of f as in Theorem 2.2. We know that

$$\|f - s_f\|_{L_\infty(\Omega)} \le K|\triangle|^3 |f|_{3,\infty,\Omega}$$

and

$$|s_f|_{3,\infty,\Omega} \le |f|_{3,\infty,\Omega} + K|\triangle|^3 |f|_{3,\infty,\Omega} = C_0 |f|_{3,\infty,\Omega},$$

where $C_0 = 1 + K|\triangle|^3$.

We recall that $Ps_f = s_f - S_f$. By Theorem 3.5,

$$|s_f - S_f|_{3,\infty,\Omega} = |Ps_f|_{3,\infty,\Omega} \le C_3 |s_f|_{3,\infty,\Omega}.$$

Since $(s_f(v) - S_f(v)) = 0$ for all vertices v of $\triangle$ and $\dfrac{\partial}{\partial x}(s_f(v_b) - S_f(v_b)) = 0$, $\dfrac{\partial}{\partial y}(s_f(v_b) - S_f(v_b)) = 0$ for all boundary vertices v_b of $\triangle$, by Lemma 3.3 and 3.4,

$$\|s_f - S_f\|_{L_\infty(\Omega)} \le C_4 |\triangle|^3 |s_f - S_f|_{3,\infty,\Omega}$$

and hence,

$$\|s_f - S_f\|_{L_\infty(\Omega)} \le C_0 C_3 C_4 |\triangle|^3 |f|_{3,\infty,\Omega}$$

where $C_4 = \max\{C_1, C_2\}$. Then the error bound (3.18) follows from

$$\|f - S_f\|_{L_\infty(\Omega)} \le \|f - s_f\|_{L_\infty(\Omega)} + \|s_f - S_f\|_{L_\infty(\Omega)}.$$

This completes the proof. ♠

4 Estimate of Boundary Derivatives

When a set of data is given, we may not have boundary derivatives. In this case, we have to estimate their values. There are many local techniques to estimate derivatives based on discrete function values in the literature. We discuss three different approaches to see how they affect on the approximation of the minimal energy spline interpolation.

4.1 Lagrange Interpolation Scheme with E_3

As we do not have derivative values at the boundary, the easiest approach is to do nothing. That is, we just simply use Lagrange interpolatory scheme with E_3 based on the spline space $S_8^2(\triangle)$ as the clamped spline interpolation. That is to find a spline $s^* \in S_8^2(\triangle)$ satisfying (1.1) such that

$$E_3(s^*) = \min\{E_3(s) : s(v_i) = z_i, i = 1, \ldots, n, s \in S_8^2(\triangle)\}.$$

The existence and uniqueness of such minimizing spline s^* are the similar to the clamped spline interpolation. We leave the proof to the interested reader. We note that one could find an $s^* \in S_5^1(\triangle)$ satisfying (1.1) which minimizes higher order energy functional E_3 instead of E_2. Our numerical experiments clearly show that this idea does not work. See Remark 6.1 in Section 6.

Example 4.1 *In this example we demonstrate the effectiveness of energy functionals $E_2(f)$ and $E_3(f)$. We use the triangulations in Figures 1 and 2, testing functions and spline spaces as in Example 3.1 while replacing energy functional $E_3(f)$ by $E_2(f)$ for the Lagrange spline interpolation using $S_8^2(\triangle)$. Tables 2(a) and 2(b), respectively give the maximum errors of clamped spline interpolation and Lagrange spline interpolation on different triangulation.*

$f_1 \setminus N$	4	8	16	32
Clamped Interp.	$5.95e-004$	$6.12e-005$	$7.56e-006$	$9.80e-007$
Lagrange use E_3	$6.01e-003$	$6.62e-004$	$8.23e-005$	$9.69e-006$
Lagrange use E_2	$6.85e-002$	$1.70e-002$	$4.21e-003$	$1.06e-003$
$f_2 \setminus N$	4	8	16	32
Clamped Interp.	$4.27e-004$	$7.37e-005$	$9.92e-006$	$6.58e-007$
Lagrange use E_3	$4.03e-003$	$4.10e-004$	$5.91e-005$	$6.47e-006$
Lagrange use E_2	$2.38e-002$	$5.51e-003$	$1.32e-003$	$3.20e-004$
$f_3 \setminus N$	4	8	16	32
Clamped Interp.	$8.13e-002$	$3.62e-002$	$7.44e-004$	$8.24e-005$
Lagrange use E_3	$1.59e-001$	$3.65e-002$	$1.25e-003$	$1.11e-004$
Lagrange use E_2	$9.28e-002$	$5.01e-002$	$4.18e-003$	$5.63e-004$

Table 2(a). Maximum errors on triangulation in Figure 1

$f_1 \ \backslash \ N$	5	9	17	33
Clamped Interp.	$8.16e-004$	$1.02e-004$	$1.14e-005$	$5.66e-006$
Lagrange use E_3	$6.92e-003$	$8.24e-004$	$8.03e-005$	$1.38e-005$
Lagrange use E_2	$6.16e-002$	$2.26e-002$	$4.04e-003$	$1.12e-003$
$f_2 \ \backslash \ N$	5	9	17	33
Clamped Interp.	$5.29e-004$	$1.32e-004$	$4.67e-005$	$9.01e-006$
Lagrange use E_3	$4.41e-003$	$4.10e-004$	$7.30e-005$	$2.11e-005$
Lagrange use E_2	$2.22e-002$	$5.31e-003$	$1.44e-003$	$3.06e-004$
$f_3 \ \backslash \ N$	5	9	17	33
Clamped Interp.	$1.21e-001$	$6.19e-002$	$7.51e-004$	$1.77e-004$
Lagrange use E_3	$1.59e-001$	$6.47e-002$	$1.43e-003$	$2.50e-004$
Lagrange use E_2	$1.19e-001$	$8.54e-002$	$4.98e-003$	$8.43e-004$

Table 2(b). Maximum errors on triangulation in Figure 2

Discussion: *From Tables 2, it is clear to see that the maximum errors of the Lagrange interpolation using* E_3 *are much better than that of using* E_2*. Thus we recommend to use* E_3 *for practical purpose.*

4.2 Least Squares Approach

One obvious approach is to use least squares estimate of derivatives based on discrete function values. The main idea is to construct the least squares polynomial of degree, say 1 from several points nearby a boundary vertex v_b and use the derivatives of the least squares polynomials for approximating derivatives at v_b. Assuming $v_1, v_2, \ldots, v_m$ are m vertices which are connected to v_b by an edge of $\triangle$. We solve

$$\min_{a,b,c} \sum_{i=1}^{m} |a + bx + cy - f(v_i)|^2$$

Then we use b for $D_x f(v_b)$ and c for $D_y f(v_b)$. Similarly, we can construct the least squares polynomial of higher order degrees. That is, let $v_1, v_2, \cdots, v_m$ are vertices of $\triangle$ which are connected to v_b by at most two edges of $\triangle$. Suppose $g(x, y)$ is the bivariate polynomial of degree 2 which solves the following minimization problem

$$\min_{c_1,c_2,\cdots,c_6} \sum_{j=1}^{m} |c_1 + c_2 x_j + c_3 y_j + c_4 x_j y_j + c_5 x_j^2 + c_6 y_j^2 - f(v_j)|^2, \qquad (4.1)$$

where $v_j = (x_j, y_j)$. Then we use the derivative of $g(x, y)$ to approximate the derivatives of f at v_b. We call these estimates the $LSd2$ method. It is easy to see the least squares method can reproduce all polynomials of degree 2. Also one can easily extend this method to improve the accuracy by adding more nearby function values and using higher order polynomials. That is, we can have $LSd3$ and $LSd4$ methods which can reproduce all polynomials of degree 3 and 4, respectively.

We will list the maximum error of estimating derivatives by least squares approach in the following subsection.

4.3 Multiple Point Approach

Let us start with a concrete example to explain the multiple point methods. Let $v_1, v_2, \ldots, v_5$ be five vertices which are connected to a boundary vertex v_b. Suppose

$$D_x f(v_b) = c_1 f(v_b) + c_2 f(v_1) + c_3 f(v_2) + c_4 f(v_3) + c_5 f(v_4) + c_6 f(v_5) \quad (4.2)$$

for some coefficients to be solved as follows. Using Taylor formula, we have

$$\begin{aligned} f(v_i) &= f(v_b) + (x_i - x_b) D_x f(v_b) + (y_i - y_b) D_y f(v_b) \\ &+ \frac{(x_i - x_b)^2}{2} D_x^2 f(v_b) + \frac{(y_i - y_b)^2}{2} D_y^2 f(v_b) \\ &+ (x_i - x_b)(y_i - y_b) D_x f(v_b) D_y f(v_b) + \mathcal{O}(h^3), \end{aligned}$$

for $i = 1, 2, \ldots, 5$. Then replacing each $f(v_i)$ in equation (4.2), we can get the following equations:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & x_1 - x_b & y_1 - y_b & \frac{(x_1-x_b)^2}{2} & \frac{(y_1-y_b)^2}{2} & (x_1 - x_b)(y_1 - y_b) \\ 1 & x_2 - x_b & y_2 - y_b & \frac{(x_2-x_b)^2}{2} & \frac{(y_2-y_b)^2}{2} & (x_2 - x_b)(y_2 - y_b) \\ 1 & x_3 - x_b & y_3 - y_b & \frac{(x_3-x_b)^2}{2} & \frac{(y_3-y_b)^2}{2} & (x_3 - x_b)(y_3 - y_b) \\ 1 & x_4 - x_b & y_4 - y_b & \frac{(x_4-x_b)^2}{2} & \frac{(y_4-y_b)^2}{2} & (x_4 - x_b)(y_4 - y_b) \\ 1 & x_5 - x_b & y_5 - y_b & \frac{(x_5-x_b)^2}{2} & \frac{(y_5-y_b)^2}{2} & (x_5 - x_b)(y_5 - y_b) \end{pmatrix}^T \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

We will assume that the matrix above is invertible. In fact, there are usually more than 5 points near to v_b and we have many choices of the linear system above. We should choose the coefficient matrix which has the smallest condition number. Solving above matrix, we can have coefficients c_i to get an approximation of $D_x f(v_b)$. Similarly we can estimate $D_y f(v_b)$ in the same way. We call this method Five Points method. It is easy to see the error term in this method is $\mathcal{O}(h^3)$.

Of course, we can increase the accuracy to $\mathcal{O}(h^4)$ by constructing the Nine Points method or $\mathcal{O}(h^5)$ by the Fourteen Points method. The details are omitted here.

Here we list the maximum errors of estimating derivatives by different methods on different triangulations in Figure 1 and 2.

f_1	4	8	16	32
LSd2 method	$2.05e-001$	$5.11e-002$	$1.28e-002$	$3.21e-003$
LSd3 method	$7.54e-015$	$1.24e-014$	$1.24e-014$	$1.59e-014$
LSd4 method	$1.86e-014$	$2.30e-014$	$1.08e-014$	$2.48e-014$
Five points	$1.25e-001$	$3.13e-002$	$7.84e-003$	$2.02e-003$
Nine points	$1.24e-014$	$2.48e-014$	$4.97e-014$	$9.94e-014$
Fourteen points	$2.48e-014$	$3.37e-014$	$9.23e-014$	$1.70e-013$
f_2	4	8	16	32
LSd2 method	1.4564	$7.24e-001$	$3.54e-001$	$1.74e-001$
LSd3 method	$5.35e-001$	$1.72e-001$	$4.68e-002$	$1.21e-002$
LSd4 method	$9.22e-001$	$7.34e-002$	$7.42e-003$	$8.19e-004$
Five points	$1.63e-001$	$4.14e-002$	$1.04e-002$	$2.63e-003$
Nine points	$5.99e-002$	$7.74e-003$	$9.74e-004$	$1.21e-004$
Fourteen points	$2.37e-002$	$1.51e-003$	$9.73e-005$	$6.09e-006$
f_3	4	8	16	32
LSd2 method	1.2907	1.8718	$3.20e-001$	$1.27e-001$
LSd3 method	4.5926	2.1971	$4.62e-001$	$5.69e-002$
LSd4 method	$1.24e+001$	1.1238	$5.42e-001$	$2.58e-002$
Five points	1.1808	1.1224	$2.50e-001$	$4.91e-002$
Nine points	3.3594	$7.41e-001$	$2.88e-001$	$1.53e-002$
Fourteen points	5.0417	$8.55e-001$	$1.12e-001$	$2.40e-002$

Table 3(a). Maximum errors of derivatives on uniform triangulation

From the Table 3(a), we can see that maximum error of the multiple point approach is better than that of least squares approach on uniform triangulation. So we recommend to use multiple point methods when dealing with uniform triangulation. In addition, it is easy to see that five, nine and fourteen points methods can reproduce derivatives of polynomials of degree 2, 3 and 4, respectively.

f_1	5	9	17	33
LSd2 method	$2.05e-001$	$5.29e-002$	$3.02e-002$	$3.41e-003$
LSd3 method	$1.77e-014$	$4.04e-014$	$1.61e-013$	$1.90e-013$
LSd4 method	$3.46e-014$	$9.90e-014$	$3.48e-013$	$2.23e-013$
Five points	$1.93e-001$	$1.10e-001$	$1.40e-001$	$9.75e-002$
Nine points	$9.23e-014$	$6.25e-013$	$2.91e-010$	$1.45e-011$
Fourteen points	$1.20e-013$	$1.36e-012$	$4.14e-012$	$8.64e-012$
f_2	5	9	17	33
LSd2 method	1.5170	$8.12e-001$	$3.54e-001$	$1.82e-001$
LSd3 method	$5.59e-001$	$2.11e-001$	$4.75e-002$	$1.31e-002$
LSd4 method	1.0029	$8.95e-002$	$7.31e-003$	$8.86e-004$
Five points	$2.41e-001$	$1.29e-001$	$2.03e-002$	$5.69e-002$
Nine points	$1.52e-001$	$3.22e-001$	$6.42e-001$	$1.46e-001$
Fourteen points	1.3586	1.9773	$1.71e-002$	$2.12e-003$
f_3	5	9	17	33
LSd2 method	1.6300	1.9581	$3.32e-001$	$1.22e-001$
LSd3 method	5.0691	2.6041	$4.90e-001$	$6.12e-002$
LSd4 method	$1.30e+001$	1.9237	$3.61e-001$	$4.54e-002$
Five points	1.3030	1.4635	$2.52e-001$	1.0468
Nine points	$1.28e+001$	$2.48e+001$	$1.68e+001$	2.1297
Fourteen points	$5.57e+001$	$2.73e+001$	5.3694	3.1966

Table 3(b). Maximum errors of derivatives on arbitrary triangulation

Table 3(b) show that each method of multiple point approach does not work well for arbitrary triangulations. So we recommend to use least squares approach when dealing with arbitrary(non-uniform) triangulation.

Having these boundary derivative values, we can clamp down the spline interpolation using E_3 as in the previous section.

Example 4.2 *We use the uniform triangulation in Figure 1, the arbitrary triangulation in Figure 2, the same testing functions and spline spaces as in Example 3.1 for each method. Table 4. lists the maximum errors of each method together with Lagrange interpolation and clamped interpolation based on energy functional* E_3.

$f_1 \ \backslash \ N$	4	8	16	32
Lagrange use E_3	$6.01e-003$	$6.62e-004$	$8.23e-005$	$9.69e-006$
Five points	$7.51e-003$	$9.05e-004$	$1.12e-004$	$1.56e-005$
Nine points	$5.95e-004$	$6.12e-005$	$7.56e-006$	$9.80e-007$
Fourteen points	$5.95e-004$	$6.12e-005$	$7.56e-006$	$9.80e-007$
Clamped Interp.	$5.95e-004$	$6.12e-005$	$7.56e-006$	$9.80e-007$
$f_2 \ \backslash \ N$	4	8	16	32
Lagrange use E_3	$4.02e-003$	$2.41e-003$	$5.91e-005$	$6.47e-006$
Five points	$7.71e-003$	$1.12e-003$	$1.44e-004$	$1.46e-005$
Nine points	$4.51e-003$	$3.06e-004$	$1.78e-005$	$5.73e-006$
Fourteen points	$1.81e-003$	$1.14e-004$	$1.12e-005$	$5.61e-006$
Clamped Interp.	$4.27e-004$	$7.37e-005$	$9.92e-006$	$6.58e-007$
$f_3 \ \backslash \ N$	4	8	16	32
Lagrange use E_3	0.1592	0.0365	0.0012	$1.11e-004$
Five points	$1.12e-001$	$3.92e-002$	$3.22e-003$	$3.02e-004$
Nine points	$1.92e-001$	$3.63e-002$	$1.12e-003$	$1.01e-004$
Fourteen points	$3.26e-001$	$3.10e-002$	$1.41e-003$	$1.11e-004$
Clamped Interp.	$8.13e-002$	$3.62e-003$	$7.44e-004$	$8.24e-005$

Table 4(a). Maximum errors for multiple point approach on unform triangulation

$f_1 \ \backslash \ N$	5	9	17	33
Lagrange use E_3	$6.91e-003$	$8.24e-004$	$8.03e-005$	$1.38e-005$
LSd2 method	$1.09e-002$	$1.41e-003$	$1.67e-004$	$2.15e-005$
LSd3 method	$8.16e-004$	$1.02e-004$	$1.14e-005$	$5.66e-006$
LSd4 method	$8.16e-004$	$1.02e-004$	$1.14e-005$	$5.66e-006$
Clamped Interp.	$8.16e-004$	$1.02e-004$	$1.14e-005$	$5.66e-006$
$f_2 \ \backslash \ N$	5	9	17	33
Lagrange use E_3	$4.42e-003$	$4.10e-004$	$7.30e-005$	$2.11e-005$
LSd2 method	$6.76e-002$	$1.69e-002$	$4.11e-003$	$9.93e-004$
LSd3 method	$2.53e-002$	$4.12e-003$	$5.90e-004$	$7.13e-005$
LSd4 method	$4.45e-002$	$1.81e-003$	$9.16e-005$	$1.89e-005$
Clamped Interp.	$5.29e-004$	$1.32e-004$	$4.67e-005$	$9.01e-006$
$f_3 \ \backslash \ N$	5	9	17	33
Lagrange use E_3	$1.59e-001$	$6.47e-002$	$1.41e-003$	$2.50e-004$
LSd2 method	$1.21e-001$	$4.32e-002$	$4.51e-003$	$7.41e-004$
LSd3 method	$3.14e-001$	$4.89e-002$	$5.81e-003$	$3.49e-004$
LSd4 method	$7.13e-001$	$3.14e-002$	$4.61e-003$	$2.05e-004$
Clamped Interp.	$1.21e-001$	$6.19e-002$	$7.51e-004$	$1.77e-004$

Table 4(b). Maximum errors for least squares approach on arbitrary triangulation

Discussion: *From Table 4(a) and 4(b), we can see that 14 point method*

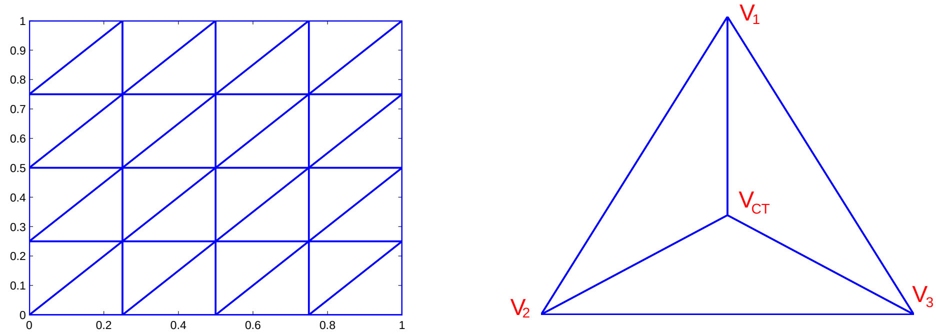


Figure 4: A given triangulation $\triangle$ (left) and CT refinement (right)

and least squares LSd4 method are better than Lagrange interpolation on uniform triangulation and arbitrary triangulation.

5 Boundary Clough-Tocher Scheme

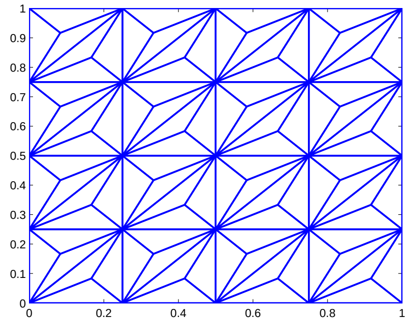
In this section we study a generalization of the univariate not-a-knot cubic spline interpolation (cf. [2]). Suppose $V = \{(x_i, y_i)\}_{i=1}^n$ is a set of data locations and $\{z_i, i = 1, \cdots, n\}$ are given real values. Given a triangulation $\triangle$ of the data locations (e.g., Fig 4 left), we first refine all the triangles in $\triangle$ by Clough-Tocher refinement (see Fig 4 right) to get a new triangulation $\triangle_{CT}$ (see Fig 5 left). Let V_{CT} be the collection of interior vertex in each triangles and V_{BCT} be the collection of interior vertex in each boundary triangles (see Fig 5 right). Here we use the following special spline space

$$\begin{aligned} S_7^{2,3,4}(\triangle_{CT}) \quad := \quad & \{s \in C^2(\triangle_{CT}) : s|_T \in P_3 \quad \forall T \in \triangle_{CT}, \\ & \text{and} \quad s \in C^3(v_{CT}), \quad \forall v_{CT} \in V_{CT}, \\ & \text{and} \quad s \in C^4(v_{BCT}) \quad \forall v_{BCT} \in V_{BCT}\}. \end{aligned}$$

Then boundary CT scheme(BCT) is to find a spline $s_* \in S_7^{2,3,4}(\triangle_{CT})$ such that

$$E_3(s_*) = \min\{E_3(s) : s(x_i, y_i) = z_i, i = 1, \ldots, n, s \in S_7^{2,3,4}(\triangle_{CT})\}.$$

Example 5.1 *In this example we present numerical results based on a boundary CT method over triangulations in Figures 1 and 2 and testing functions as in Example 4.1. Table 5. lists the maximum errors of this method and Lagrange interpolation method using E_3 as we know from a previous section*



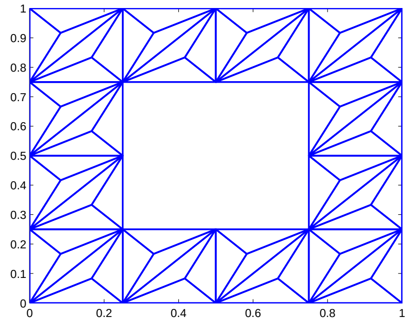


Figure 5: Triangulation $\triangle_{CT}$ (left) and boundary triangles in $\triangle_{CT}$ (right)

that E_3 is better than E_2. Table 6. lists the approximation order of this method. Here we use $S_8^2(\triangle)$ for Lagrange interpolation on triangulations of data locations in Figures 1 and 2.

$f_1 \setminus N$	4	8	16	32
BCT Scheme	$6.01e-003$	$6.43e-004$	$4.93e-005$	$4.96e-006$
Lagrange Int. with E_3	$6.01e-003$	$6.62e-004$	$8.23e-005$	$9.69e-006$
$f_2 \setminus N$	4	8	16	32
BCT Scheme	$4.02e-003$	$3.93e-004$	$2.97e-005$	$6.32e-006$
Lagrange Int. with E_3	$4.02e-003$	$4.10e-004$	$5.91e-005$	$6.47e-006$
$f_3 \setminus N$	4	8	16	32
BCT Scheme	$1.59e-001$	$3.62e-002$	$2.37e-004$	$7.37e-005$
Lagrange Int. with E_3	$8.13e-002$	$3.62e-002$	$7.44e-004$	$8.24e-005$

Table 5(a). Maximum errors on triangulations in Figure 1

$f_1 \setminus N$	5	9	17	33
BCT Scheme	$5.21e-003$	$3.15e-004$	$5.56e-005$	$1.15e-005$
Lagrange Int. with E_3	$6.91e-003$	$8.24e-004$	$8.03e-005$	$1.38e-005$
$f_2 \setminus N$	5	9	17	33
BCT Scheme	$3.21e-003$	$1.49e-004$	$4.33e-005$	$1.08e-005$
Lagrange Int. with E_3	$4.41e-003$	$6.70e-004$	$7.30e-005$	$2.11e-005$
$f_3 \setminus N$	5	9	17	33
BCT Scheme	$3.80e-001$	$5.47e-002$	$1.31e-003$	$1.56e-004$
Lagrange Int. with E_3	$1.59e-001$	$6.47e-002$	$1.41e-003$	$2.50e-004$

Table 5(b). Maximum errors on triangulations in Figure 2

Discussion: *From Tables 5(a) and 5(b), we can see the maximum errors of boundary CT scheme are slightly better than that of Lagrange interpolation using* E_3*. Thus we recommend the boundary CT scheme.*

f_1	9.33	13.04	9.94
f_2	10.18	13.23	4.70
f_3	4.40	15.27	3.22

Table 6(a). Approximation order on triangulations in Figure 1

f_1	16.51	5.67	4.83
f_2	21.48	3.44	4.01
f_3	6.95	42.08	8.33

Table 6(b). Approximation order on triangulations in Figure 2

Discussion:*From Table 6(a) and 6(b), we can see the approximation order is higher than* 4 *which is what we expect.*

Certainly, one can use other refinement method, e.g., the Powell-Sabin refinements and even uniform refinement technique to replace the Clough-Tocher method for constructing other interpolatory schemes with higher order of approximation. We leave it to the interested readers.

6 Remark

We have some remarks in order.

Remark 6.1 *According to the definition of* E_3*, it requires the* 3*th derivatives of bivariate spline. So it is necessary to use at least* C^2 *spline functions and only* C^1 *splines are not enough. Let us present a table of maximum errors to show that Lagrange interpolation using* E_3 *based on spline space* S_5^1 *on the same triangulations as in Fig. 1. From the table, we can conclude that the* C^1 *quintic interpolatory splines do not converge at all. This shows that a* C^2 *spline space using the energy functional* E_3 *is necessary.*

$f \setminus N$	4	8	16	32
f_1	8.1745	3.8420	3.8144	55.7165
f_2	0.3695	0.2639	0.0944	0.7254
f_3	1.1835	0.3454	0.1937	2.8947

Remark 6.2 *We list the approximation orders for various test functions of each scheme on the triangulations in Fig. 1. The approximation orders are computed by dividing maximum errors by the size of two consecutive triangulations. We must point out that the approximation orders are different from each other since the complexity of test functions are different.*

f_1	*Lagrange using* E_2	4.02	4.04	4.20
	Lagrange using E_3	9.64	7.55	8.49
	Hermite Interpolation	8.01	8.05	10.44
	Clamped Interpolation	9.72	8.09	7.71
	BCT Scheme	9.33	13.04	9.93
f_2	*Lagrange using* E_2	4.49	4.07	4.06
	Lagrange using E_3	1.66	40.06	9.13
	Hermite Interpolation	8.69	8.53	11.56
	Clamped Interpolation	5.79	7.42	15.07
	BCT Scheme	10.17	13.23	4.69
f_3	*Lagrange using* E_2	1.85	12.21	7.28
	Lagrange using E_3	4.36	30.14	10.90
	Hermite Interpolation	5.46	37.07	3.07
	Clamped Interpolation	2.24	48.65	9.02
	BCT Scheme	4.40	152.74	3.21

Table 7: Approximation rates for various schemes

From the table 7, we can see that the approximation rates of Lagrange interpolation (using E_2*) are almost* $4 = 2^2$ *which means the approximation order is* 2*. Note that the approximation rates of Hermite and clamped interpolation are almost* $8 = 2^3$ *which confirmed our approximation theorem 3.6. And it is easy to see the approximation rates of other schemes are bigger than* 4 *which means each scheme has a higher order of approximation.*

Remark 6.3 *If a function is complicated, it will be hard to approximate. In table 8, we list the maximum errors of* $\sin(5\pi(x^2 + y^2))$ *using each scheme on triangulation (a) (See Fig. 1).*

$\sin(5\pi(x^2 + y^2)) \ \backslash \ N$	4	8	16	32
Lagrange using E_2	2.0883	2.3063	0.2309	0.0623
Lagrange using E_3	2.0143	0.9771	0.0297	0.0073
Hermite Interpolation	1.8952	0.2505	0.0196	0.0038
Clamped Interpolation	2.1483	0.8078	0.0248	0.0051
BCT Scheme	2.1967	1.3477	0.2124	0.0461

Table 8: Maximum errors for various schemes

Remark 6.4 *We could test the approximation order of interpolatory splines using a higher order energy functional* E_4*. Then it will require spline space* S_{11}^3 *and the dimension of the problems increases quickly. We are not able to compute an interpolatory spline when refining triangulations 4 times.*

References

[1] Awanou, G., M. J. Lai, and P. Wenston, The multivariate spline method for scattered data fitting and numerical solution of partial differential equations, in Wavelets and Splines: Athens 2005, G. Chen and M.-J. Lai (eds), Nashboro Press, Brentwood, 2006, 24–74.

[2] Boor, C. de, *A Practical Guide to Splines*, Springer (New York), 1978.

[3] K. Farmer and M. J. Lai, Scattered data interpolation by C^2 quintic splines using energy minimization, in: C.K. Chui and L.L. Schumaker,(Eds.), Approximation Theory IX, 1998, 47-54.

[4] M. von Golitschek, M. J. Lai and L. L. Schumaker, Error bounds for minimal energy bivariate polynomial splines, Numer. Math. **93** (2002), 315–331.

[5] M. J. Lai and L. L. Schumaker, Scattered data interpolation using C^2 supersplines of degree six, SIAM J. Numer. Anal. **34** (1997), 905-921.

[6] M. J. Lai and L. L. Schumaker, On the approximation power of bivariate splines, Adv. in Comp. Math. **9** (1998), 251-279.

[7] M. J. Lai and L. L. Schumaker, On the approximation power of splines on triangulated quadrangulations, SIAM J. Numer. Anal. **36** (1998), 143-159.

[8] M. J. Lai and L. L. Schumaker, Macro-elements and stable local bases for splines on Clough-Tocher triangulations, Numer. Math. **88** (2001), 105-119.

[9] M. J. Lai and L. L. Schumaker, Quadrilateral macro-elements, SIAM J. Math. Anal. **33** (2002), 1107-1116.

[10] M. J. Lai and L. L. Schumaker, Macro-elements and stable local bases for splines on Powell-Sabin triangulations, Math. Comput. **72** (2003), 335-354.

[11] M. J. Lai and L. L. Schumaker, *Spline Functions on Triangulations*, Cambridge University Press, 2007.

[12] T. Zhou, D. Han and M. J. Lai, Energy Minimization Method for Scattered Data Hermite Interpolation, Appl. Numer. Math. **58** (2008), 646-659.

[13] Nürnberger G., Zeilfelder F., Lagrange interpolation by bivariate C^1 splines with optimal approximation order, Advances in Comp. Math. **21** (2004), 381-419.

[14] Nürnberger G.,Rayevskaya V., Schumaker L.L., Zeilfelder F., Local Lagrange interpolation with bivariate splines of arbitrary smoothness, Constr. Approx. **23** (2006), 33-59.

[15] Matt, A.M., Nürnberger G., Local Lagrange interpolation using cubic C^1 splines on type-4 cube partitions, J. Approx. Theory, doi: 10.1016/j.jat.2009.07.009.