



# Trivariate Biharmonic B-Splines

Fei Hou<sup>1</sup>, Hong Qin<sup>2</sup> and Aimin Hao<sup>1</sup>

<sup>1</sup>State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China  
houfei@vrlab.buaa.edu.cn, ham@buaa.edu.cn

<sup>2</sup>Stony Brook University, NY, USA  
qin@cs.sunysb.edu

---

## Abstract

*In this paper, we formulate a novel trivariate biharmonic B-spline defined over bounded volumetric domain. The properties of bi-Laplacian have been well investigated, but the straightforward generalization from bivariate case to trivariate one gives rise to unsatisfactory discretization, due to the dramatically uneven distribution of neighbouring knots in 3D. To ameliorate, our original idea is to extend the bivariate biharmonic B-spline to the trivariate one with novel formulations based on quadratic programming, approximating the properties of localization and partition of unity. And we design a novel discrete biharmonic operator which is optimized more robustly for a specific set of functions for unevenly sampled knots compared with previous methods. Our experiments demonstrate that our 3D discrete biharmonic operators are robust for unevenly distributed knots and illustrate that our algorithm is superior to previous algorithms.*

**Keywords:** biharmonic b-spline, green's function, quadratic programming

**ACM CCS:** I.3.5 [Computer Graphics]: Computational Geometry and Object Modelling—Splines

---

## 1. Introduction and Motivation

Data fitting/interpolation is a fundamental and enabling computational method in a wide variety of computer graphics subfields including geometric modelling, image/shape/video processing, scientific visualization, etc. It is also ubiquitous spanning across many disciplines in science, engineering, medicine, etc. One of the most powerful technical approaches is to interpolate/approximate discrete observations using certain types of basis functions towards continuous representations in the interest of accuracy and compactness.

B-spline is one such family of well-investigated basis functions with many powerful and attractive properties that have warranted their widespread penetration into the above-documented fields. In the univariate case, divided differences are relevant to the construction of B-splines. Feng and Warren [FW12] generalized the divided difference to 2D planar (Euclidean) domain and 2D manifold based on discrete Laplacian, and then extended them to discrete bi-Laplacian to define 2D biharmonic B-splines. However, the behaviours/properties of biharmonic B-splines and their construction in 3D domain are far from trivial, still remaining as one of the open research problems. Their direct generalization to 3D domain

may give rise to seriously distorted basis functions when the sample knots are uneven. The theoretical results and their subsequent comparisons elaborated in this paper showcase the first attempt on trivariate biharmonic B-splines. While defining irregular knots over arbitrary volumetric grids, we systematically study the properties of 3D biharmonic B-splines and devise a novel function-specific discrete bi-Laplacian operator in a bounded 3D Euclidean domain.

Our key motivation is to generalize the bivariate biharmonic B-spline to trivariate biharmonic B-spline to broaden its application scopes. The conventional discrete bi-Laplacian is usually evaluated by iterated discrete Laplacian. According to Feng and Warren [FW12], it is unstable on irregular grid. They have improved the iterated Laplacian and presented a cubic precision discrete bi-Laplacian on unbounded 2D domain. Nevertheless, the distribution of neighbouring knots in 3D is much more complicated than its arrangement in any 2D setting, which may lead to seriously distorted basis functions when the sample knots are uneven. One originality of this paper is our newly designed and well-thought method to compute the function-specific discrete bi-Laplacian operator via an optimization formulation, which still approximates the properties of localization and partition of unity

simultaneously. The first step of our novel approach is to compute the discrete Laplacian operator locally for each knot, followed by the computation of the discrete bi-Laplacian operator for all the knots simultaneously in the succeeding step. Both of them are formulated as quadratic programming problems.

The salient contributions of our paper include:

- (1) With mathematical rigor, we generalize the bivariate biharmonic B-spline to trivariate biharmonic B-spline and prove that they are localized and approximate a partition of unity. To our best knowledge, it is the first attempt trying to tackle the underlying technical challenges inherent in trivariate biharmonic B-splines.
- (2) We propose an improved formulation and design a novel discrete trivariate Laplacian as well as bi-Laplacian operator optimized for a specific set of functions, which is more robust than previous discrete operators for unevenly distributed sample knots.
- (3) We analyse the stability and robustness of our new formulation, and conduct comparison to the previous method to account for the superiority of our algorithm.

## 2. Related Work

**Basis Function and Function Fitting.** Radial basis function is one family of the most widely used basis functions consisting of various categories of functions [Buh03], such as Gaussian kernel, etc. In such cases, the basis functions at different knots are independent of each other, so they would not form a partition of unity. Spline functions, such as the thin-plate spline [Boo89], T-spline [SZBN03], polycube splines [WHL\*08] and Voronoi splines [ME10], are a large and diverse family of basis functions. Feng and Warren [FW12] make use of the connection between divided difference and discrete Laplacian and present biharmonic B-spline on 2D domains, which is localized and forms a partition of unity. But the trivial generalization to 3D domain is insufficient, since the complex neighbouring structure of knots leads to unstable behaviours. Moreover, the behaviour of 3D biharmonic basis is still unclear. In this paper, we investigate the properties of 3D biharmonic basis on bounded domain and present a novel discrete operator to evaluate the basis robustly, which does not rely on the Voronoi cells either.

**Harmonic and Biharmonic Basis.** Harmonic and biharmonic operators play a fundamental role in computer graphics and they are widely used in different applications, such as heat kernel signature [SOG09], biharmonic distance [LRF10] and spectral graph [Chu97], etc. In geometry processing, the discrete Laplacian operator has been deeply studied in previously published work. Meyer *et al.* [MDSB03] proposed cotangent weight discrete Laplacian operator for triangulated two-manifolds. Hildebrandt and Polthier [HP11] constructed strongly consistent discrete Laplacian–Beltrami operators on polyhedral surface. Wardetzky *et al.* [WMKG07] reviewed the discrete Laplacian operators. Many methods use iterated Laplacian for discrete bi-Laplacian [JTSZ10, LRF10]. Feng and Warren [FW12] proposed to enhance the results by making use of additional constraints to improve them to cubic precision. But, we find that it is still inadequate for 3D uneven knots.

In sharp contrast to previous methods independent of the differential function, we present a function-specific discrete bi-Laplacian operator which is optimized for the specified set of functions.

**Interpolation.** The interpolation problem has been widely studied in different areas. Many methods have been proposed to interpolate values from scattered points [Lev98, OBA\*03, Wen05]. Ern and Guermond [EG04] introduced interpolation using finite elements. Ju *et al.* [JSW05] proposed mean value coordinate to interpolate the values smoothly on closed triangular meshes and Li *et al.* [LJH13] proposed cubic mean value coordinate to interpolate both boundary value and gradients over a 2D polygonal domain. Weber *et al.* [WPG12] presented a new type of barycentric coordinate named biharmonic coordinate which is natural generalization of harmonic coordinate [JMD\*07] and is useful for various applications, such as data interpolation and image deformation. Finch *et al.* [FSH11] interpolated data for vector graphics using the thin-plate spline which is expanded by Green’s functions. In this paper, we define the biharmonic B-spline basis as a combination of Green’s functions to approximate the integral of  $\delta$  function.

## 3. Brief Review of Bivariate Unbounded Biharmonic B-Splines

According to [FW12], the bivariate harmonic B-spline is constructed from Green’s functions which are the solution of the harmonic equation

$$\Delta\phi_y(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{y}), \tag{1}$$

where  $\Delta$  is the Laplacian operator and  $\delta$  is the Dirac delta function. On a 2D plane, the solution of Equation (1) is a Green’s function  $\phi_y(\mathbf{x}) = \frac{1}{2\pi} \log(\|\mathbf{x} - \mathbf{y}\|)$ . Given a set of centres  $\mathbf{t}_0, \dots, \mathbf{t}_n$  (points on the 2D plane), a B-spline function  $\psi(\mathbf{x})$  can be constructed as a linear combination of the translated Green’s functions centred at each knot  $\mathbf{t}_i$  (i.e. values of different functions at a common point),  $\psi(\mathbf{x}) = \sum_i n_i \phi_{\mathbf{t}_i}(\mathbf{x})$ . Because of the symmetry of the Green’s function,  $\psi(\mathbf{x}) = \sum_i n_i \phi_{\mathbf{x}}(\mathbf{t}_i)$ . Therefore, the  $\psi(\mathbf{x})$  can also be regarded as a linear combination of a Green’s function evaluated at different points  $\mathbf{t}_i$  (i.e. values of a function at different points), which is capable of representing a discrete differential operator of the Green’s function. If the mask  $n_i$  approximates the discrete Laplacian operator,  $\psi(\mathbf{x})$  approximates the behaviour of Dirac delta function [FW12].

In a closed region  $\Omega$ , the Green’s theorem states that

$$\int \int_{\mathbf{x} \in \Omega} \Delta\phi_y(\mathbf{x}) dA = \oint_{\mathbf{x} \in \partial\Omega} \frac{\partial\phi_y(\mathbf{x})}{\partial\mathbf{n}} ds, \tag{2}$$

where  $\mathbf{n}$  is the outward unit normal vector on the boundary  $\partial\Omega$  of  $\Omega$ . The left-hand side of Equation (2) equals to 1 if point  $\mathbf{y}$  is in  $\Omega$ , and otherwise it equals to 0. Supported by Voronoi tessellation, the discrete line integral of function  $\phi_{\mathbf{x}}(\mathbf{y})$  is formulated as

$$\sum_{\mathbf{t}_i \in \mathcal{N}_j} \frac{\|\mathbf{v}_{ij}\|}{\|\mathbf{e}_{ij}\|} (\phi_{\mathbf{x}}(\mathbf{t}_i) - \phi_{\mathbf{x}}(\mathbf{t}_j)) \simeq \oint_{\mathbf{y} \in \partial\mathcal{C}_j} \frac{\partial\phi_{\mathbf{x}}(\mathbf{y})}{\partial\mathbf{n}} ds, \tag{3}$$

where  $\mathbf{t}_j$  is the  $j$ th knot and  $\mathcal{N}_j$  is the one-ring neighbouring (adjacent) knot set of  $\mathbf{t}_j$  (excluding  $\mathbf{t}_j$ ).  $\mathcal{C}_j$  is the Voronoi cell of knot  $\mathbf{t}_j$ .  $\mathbf{v}_{ij}$  is the edge between knots  $\mathbf{t}_i$  and  $\mathbf{t}_j$  and  $\mathbf{e}_{ij}$  is the Voronoi edge perpendicular to edge  $\mathbf{v}_{ij}$ . The harmonic B-spline is defined by the discrete integral in Equation (3), satisfying the form of  $\psi(\mathbf{x}) = \sum_i n_i \phi_{\mathbf{t}_i}(\mathbf{x})$ . The harmonic B-spline is provably localized and forms a partition of unity.

Analogously, the biharmonic equation is  $\Delta^2 \phi_{\mathbf{y}}(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{y})$  whose solution is  $\phi_{\mathbf{y}}(\mathbf{x}) = \frac{1}{8\pi} \|\mathbf{x} - \mathbf{y}\|^2 \log(\|\mathbf{x} - \mathbf{y}\|)$  on 2D plane. The biharmonic form of Green's theorem is

$$\int \int_{\mathbf{x} \in \Omega} \Delta^2 \phi_{\mathbf{y}}(\mathbf{x}) dA = \oint_{\mathbf{x} \in \partial\Omega} \frac{\partial \Delta \phi_{\mathbf{y}}(\mathbf{x})}{\partial \mathbf{n}} ds. \quad (4)$$

The discrete bi-Laplacian operator is generated by iterative discrete Laplacian and can be refined to cubic precision using a Vandermonde matrix based on pseudo-inverse optimization.

## 4. Trivariate Biharmonic B-Splines

### 4.1. Continuous formulation

In 3D Euclidean space, the divergence theorem states that

$$\iiint_{\Omega} (\nabla \cdot \mathbf{F}) dV = \iint_{\partial\Omega} \mathbf{F} \cdot \mathbf{n} dS. \quad (5)$$

We substitute the vector-valued function  $\mathbf{F}$  by  $\nabla \phi_{\mathbf{y}}(\mathbf{x})$ , and get

$$\iiint_{\mathbf{x} \in \Omega} \nabla^2 \phi_{\mathbf{y}}(\mathbf{x}) dV = \iint_{\mathbf{x} \in \partial\Omega} \nabla \phi_{\mathbf{y}}(\mathbf{x}) \cdot \mathbf{n} dS, \quad (6)$$

and thus,

$$\iiint_{\mathbf{x} \in \Omega} \Delta \phi_{\mathbf{y}}(\mathbf{x}) dV = \iint_{\mathbf{x} \in \partial\Omega} \frac{\partial \phi_{\mathbf{y}}(\mathbf{x})}{\partial \mathbf{n}} dS. \quad (7)$$

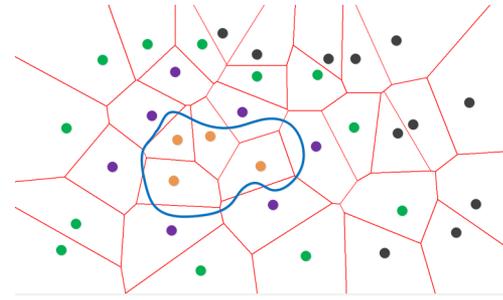
Furthermore, if the function  $\mathbf{F}$  of Equation (5) is substituted by  $\nabla(\Delta \phi_{\mathbf{y}}(\mathbf{x}))$ , we get

$$\iiint_{\mathbf{x} \in \Omega} \Delta^2 \phi_{\mathbf{y}}(\mathbf{x}) dV = \iint_{\mathbf{x} \in \partial\Omega} \frac{\partial \Delta \phi_{\mathbf{y}}(\mathbf{x})}{\partial \mathbf{n}} dS. \quad (8)$$

That is to say, the volume integral of the bi-Laplacian of  $\phi_{\mathbf{y}}(\mathbf{x})$  equals to the surface integral of the directional derivative of  $\Delta \phi_{\mathbf{y}}(\mathbf{x})$ . The integral on the right-hand side of Equation (8) with variable  $\mathbf{y}$  is called the *characteristic function*, which is a continuous form of the trivariate biharmonic basis and we will approximate it in a discrete fashion.

The solution, which is a Green's function, of biharmonic equation  $\Delta^2 \phi_{\mathbf{y}}(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{y})$  in 3D Euclidean space is  $\phi_{\mathbf{y}}(\mathbf{x}) = -\frac{1}{8\pi} \|\mathbf{x} - \mathbf{y}\|$ , where the  $-\frac{1}{8\pi}$  is the normalization factor to make the integral on a unit sphere equal to 1 if  $\mathbf{y} \in \Omega$ , and otherwise it equals 0.

As the point  $\mathbf{y}$  is moving far away from region  $\Omega$ , the characteristic function vanishes, so it is localized. Given a set of regions



**Figure 1:** 2D illustration of sample knots. The knots in set  $\mathcal{T}$  are drawn in orange bounded by blue region boundary. Their one-ring neighbours are drawn in purple and two-ring neighbours are drawn in green.  $\mathcal{T}'$  is the union of orange knots and purple knots. The black knots are discarded.

$\Omega_1, \dots, \Omega_n$ , such that  $\Omega_i \cap \Omega_j = \Phi$ ,  $i \neq j$  and  $\Omega = \cup_{i=1}^n \Omega_i$ , the sum of the characteristic functions equals to one for any fixed point  $\mathbf{y} \in \Omega$ , and otherwise it equals to zero for any fixed point  $\mathbf{y} \notin \Omega$ . Thus the sum of the characteristic functions forms a partition of unity.

### 4.2. Ghost Knots

Given a bounded 3D region  $\Omega$  and 3D knots  $\mathcal{T} = \{\mathbf{t}_1, \dots, \mathbf{t}_n\} \subset \Omega$ , we define their adjacent relationships according to their Voronoi diagram. If the Voronoi cells  $\mathcal{C}_i, \mathcal{C}_j$  have faces in common, knots  $\mathbf{t}_i$  and  $\mathbf{t}_j$  are adjacent. A knot  $\mathbf{t}_i$  is a boundary knot if its Voronoi cell is intersecting with the region boundary. Many differential operators always rely on neighbouring knots to discretize them. For instance, the discrete Laplacian operator is evaluated in the one-ring neighbourhood of the central knot. However, due to the lacking of neighbours of boundary knots, the accuracy of the discrete differential operator decreases dramatically on the region boundary. To ameliorate, we introduce additional ghost knots to assist the discretization.

We sample more ghost knots outside the region  $\Omega$  and compute the Voronoi diagram of all the knots using Voro++ [Ryc09]. Then we compute the union  $\mathcal{T}'$  of  $\mathcal{T}$  and their one-ring neighbours. If  $\mathcal{T}'$  contains boundary knots, we sample more ghost knots and recompute their Voronoi cells and  $\mathcal{T}'$  until no boundary knots are contained in  $\mathcal{T}'$ . Figure 1 shows an 2D example where the knots in  $\mathcal{T}$  are drawn in orange and one-ring and two-ring neighbouring knots are drawn in purple and green, respectively. The set  $\mathcal{T}'$  is the union of orange and purple knots. The black knots are located neither in one-ring nor two-ring neighbours so they are discarded. The purple and green knots are ghost knots. Our algorithm does not rely on the Voronoi cells explicitly but only the neighbouring relationships derived from the Voronoi diagram.

### 4.3. Discrete Laplacian Operator

**Conventional Discrete Laplacian.** With the help of ghost knots, we can discretize the characteristic function. In the Euclidean space,

given a function  $p(\mathbf{x})$ , the conventional discrete Laplacian operator is

$$(\Delta p)(\mathbf{t}_i) = \sum_{\mathbf{t}_j \in \mathcal{N}_i} (p(\mathbf{t}_j) - p(\mathbf{t}_i)). \quad (9)$$

Alternately, we have

$$\begin{aligned} \Delta p(\mathbf{x}) &= \nabla \cdot \nabla p(\mathbf{x}) = \lim_{V \rightarrow 0} \frac{1}{V} \iint_{\partial C} \nabla p \cdot \mathbf{n} \, dS \\ &= \lim_{V \rightarrow 0} \frac{1}{V} \iint_{\partial C} \frac{\partial p}{\partial \mathbf{n}} \, dS. \end{aligned} \quad (10)$$

Thus, the Laplacian operator can be discretized into

$$(\Delta p)(\mathbf{t}_i) = \frac{1}{V_i} \sum_{\mathbf{t}_j \in \mathcal{N}_i} S_j \frac{p(\mathbf{t}_j) - p(\mathbf{t}_i)}{\|\mathbf{t}_j - \mathbf{t}_i\|}, \quad (11)$$

where  $V_i$  is the volume of Voronoi cell  $\mathcal{C}_i$  and  $S_j$  is the area of the common face of Voronoi cells  $\mathcal{C}_i$  and  $\mathcal{C}_j$ . The second discretization is more accurate than the first one, but it relies on the Voronoi cell areas and volumes, which are ambiguous in the boundary Voronoi cells.

**Our Discrete Laplacian.** We present a new method to compute the discrete Laplacian operator optimized for a specific set of functions. The common properties of different discrete Laplacian operators (e.g. Equations 9 and 11) are that the two coefficients of the terms in the summation are opposite to each other. So all the coefficients' summation vanishes, and the coefficient of the central knot is negative while others are positive, which we call the intrinsic property of Laplacian operator. The Laplacian of function  $p(\mathbf{x})$  at knot  $\mathbf{t}_j$  is discretized into the form of

$$h_j p(\mathbf{t}_j) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} h_k p(\mathbf{t}_k), \quad (12)$$

where  $h_j$  denotes the  $j$ th element of the to-be-determined coefficient vector  $\mathbf{h}^{(j)}$ . Equation (12) represents the sum of differences between the central knot and all its neighbours. To evaluate the discrete Laplacian operator for the function set  $\mathcal{P} = \{p_i(\mathbf{x})\}$  at knot  $\mathbf{t}_j$ , we solve the following quadratic programming problem subject to constraints of the intrinsic properties of Laplacian operator

$$\begin{aligned} \mathbf{h}^{(j)} &= \arg \min_{\mathbf{h}^{(j)}} \sum_{p_i(\mathbf{x}) \in \mathcal{P}} (h_j p_i(\mathbf{t}_j) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} h_k p_i(\mathbf{t}_k) - \Delta p_i(\mathbf{t}_j))^2, \\ \text{s.t.} \quad & h_j + \sum_{\mathbf{t}_k \in \mathcal{N}_j} h_k = 0, \quad h_j < 0, \quad h_k \geq 0, \end{aligned} \quad (13)$$

where  $\mathbf{h}^{(j)}$  is a sparse vector whose elements uninvolved in Equation (13) are all zero.

In the framework of trivariate biharmonic B-splines, the function set is an infinite set of Green's functions  $\mathcal{P} = \{\phi_{\mathbf{t}_i}(\mathbf{x}) \mid \mathbf{t}_i \in \Omega\}$ . We sample a finite subset of functions  $\{\phi_{\mathbf{t}_i}(\mathbf{x}) \mid \mathbf{t}_i \in \mathcal{T}\}$  for optimization. In other words, we translate the Green's function to

all the knots in  $\mathcal{T}'$ . From continuous differential, we know that  $\Delta \phi_{\mathbf{t}_i}(\mathbf{x}) = -1/(4\pi \|\mathbf{x} - \mathbf{t}_i\|)$ . For the specific function set, we solve the following quadratic programming problem:

$$\begin{aligned} \mathbf{h}^{(j)} &= \arg \min_{\mathbf{h}^{(j)}} \sum_{i=1}^{|\mathcal{T}'|} \left( h_j \phi_{\mathbf{t}_i}(\mathbf{t}_j) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} h_k \phi_{\mathbf{t}_i}(\mathbf{t}_k) \right. \\ &\quad \left. - \frac{1}{|\mathcal{N}_j|} \sum_{\mathbf{t}_k \in \mathcal{N}_j} \Delta \phi_{\mathbf{t}_i} \left( \frac{\mathbf{t}_k + \mathbf{t}_j}{2} \right) \right)^2, \\ \text{s.t.} \quad & h_j + \sum_{\mathbf{t}_k \in \mathcal{N}_j} h_k = 0, \quad h_j < 0, \quad h_k \geq 0, \end{aligned} \quad (14)$$

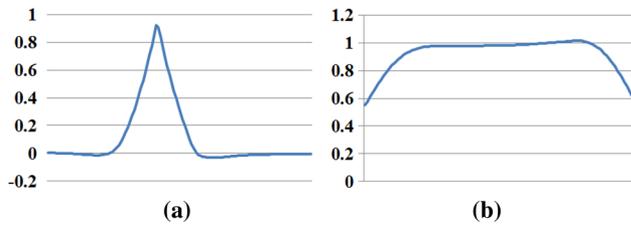
where  $|\mathcal{T}'|$  denotes the number of elements in set  $\mathcal{T}'$ . To avoid degeneration when knot  $\mathbf{t}_i$  coincides with knot  $\mathbf{t}_j$ , we use the average of Laplacians around knot  $\mathbf{t}_j$  (i.e.  $\frac{1}{|\mathcal{N}_j|} \sum_{\mathbf{t}_k \in \mathcal{N}_j} \Delta \phi_{\mathbf{t}_i}(\frac{\mathbf{t}_k + \mathbf{t}_j}{2})$ ) instead of the Laplacian at  $\mathbf{t}_j$  (i.e.  $\Delta \phi_{\mathbf{t}_i}(\mathbf{t}_j)$ ). For every knot  $\mathbf{t}_j \in \mathcal{T}'$ , we compute the discrete Laplacian operator  $\mathbf{h}^{(j)}$  using Equation (14). The functions set specific Laplacian operators minimize the error of discrete Laplacian operators as well as holding the properties of Laplacian operator.

#### 4.4. Discrete Biharmonic B-splines

The discrete trivariate biharmonic B-spline basis is to mimic the surface integral of the characteristic function. We develop a novel method to discretize the biharmonic operator. Similar to the Laplacian operator, it is also a surface integral but without being divided by the volume. Except for the intrinsic property (coefficients' summation vanishes), an important constraint is that the directional derivatives on the common face of adjacent Voronoi cells are opposite to each other since only their directions are opposite. Feng and Warren [FW12] improved the iterated discrete Laplacian and presented a cubic precision bi-Laplacian operator on 2D plane. To do so, however, they broke the intrinsic property. To evaluate the biharmonic basis for function set  $\mathcal{P} = \{p_i(\mathbf{x})\}$ , we solve the following quadratic programming:

$$\begin{aligned} \mathbf{H} &= \arg \min_{\mathbf{H}} \sum_{p_i(\mathbf{x}) \in \mathcal{P}} \sum_{j=1}^{|\mathcal{T}|} \left( \mathbf{H}_{jj} \Delta p_i(\mathbf{t}_j) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} \Delta p_i(\mathbf{t}_k) \right. \\ &\quad \left. - \iint_{\partial \mathcal{C}_j} \frac{\partial \Delta p_i(\mathbf{x})}{\partial \mathbf{n}} \, dS \right)^2, \\ \text{s.t.} \quad & \mathbf{H}_{jj} + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} = 0, \quad \mathbf{H}_{jj} < 0, \quad \mathbf{H}_{kj} \geq 0, \quad 1 \leq j \leq |\mathcal{T}|, \\ & \mathbf{H}_{kj} = \mathbf{H}_{jk}, \quad 1 \leq j \leq |\mathcal{T}|, \quad \mathbf{t}_k \in \mathcal{N}_j, \end{aligned} \quad (15)$$

where  $\mathbf{H}$  is a matrix whose  $j$ th column is the coefficient of the biharmonic basis in the region. Other uninvolved elements in Equation (15) are all zero making  $\mathbf{H}$  be sparse. The constraint  $\mathbf{H}_{kj} = \mathbf{H}_{jk}$  is necessary to make the directional derivatives on the common face of adjacent Voronoi cells be opposite to each other. Given a function  $p(\mathbf{x})$ , its bi-Laplacian basis in the region of knot  $\mathbf{t}_j$  is formulated as  $\mathbf{H}_{jj} \Delta p(\mathbf{t}_j) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} \Delta p(\mathbf{t}_k)$ .



**Figure 2:** Illustration of localization and partition of unity by the function values on a sample line. (a) Localization: the basis function values decay to zero approximately. (b) Partition of unity: the sum of basis values approximates one.

For the infinite Green's function set of trivariate B-splines, we sample a finite subset  $\{\phi_{\mathbf{t}_i}(\mathbf{x}) \mid \mathbf{t}_i \in \mathcal{T}\}$  and optimize

$$\mathbf{H} = \arg \min_{\mathbf{H}} \sum_{i=1}^{|\mathcal{T}|} \sum_{j=1}^{|\mathcal{T}|} \left( \mathbf{H}_{jj} \Delta \phi_{\mathbf{t}_i}(\mathbf{t}_j) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_i}(\mathbf{t}_k) - \delta_{ij} \right)^2, \quad (16)$$

s.t.  $\mathbf{H}_{jj} + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} = 0$ ,  $\mathbf{H}_{jj} < 0$ ,  $\mathbf{H}_{kj} \geq 0$ ,  $1 \leq j \leq |\mathcal{T}|$ ,

$$\mathbf{H}_{kj} = \mathbf{H}_{jk}, \quad 1 \leq j \leq |\mathcal{T}|, \quad \mathbf{t}_k \in \mathcal{N}_j,$$

where  $\delta_{ij} = 1$  if  $i = j$  and otherwise  $\delta_{ij} = 0$ . The Laplacian  $\Delta \phi_{\mathbf{t}_i}(\mathbf{t}_j)$  and  $\Delta \phi_{\mathbf{t}_i}(\mathbf{t}_k)$  are replaced by their discrete Laplacian operators computed in Equation (14).

We formulate the biharmonic B-spline basis function as a linear combination of Green's functions of biharmonic operator, which is similar to the thin-plate spline in spirit. Instead of interpolating control knots, we fit the values of the control knots subject to certain constraints in order to mimic the behaviour of bi-Laplacian operator.

## 5. Properties of Trivariate B-splines

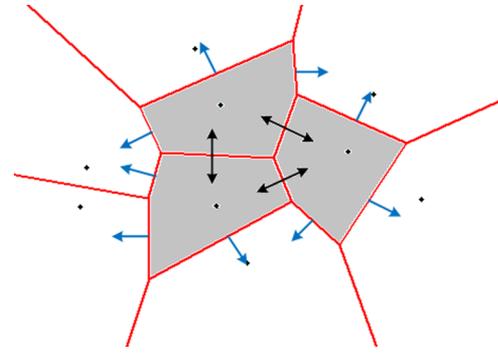
### 5.1. Localization

The discrete trivariate B-spline would not vanish exactly outside the integral region, but it decays fast as the point is moving far away from the central knot as shown in Figure 2(a).

**Theorem 1.** If the discrete biharmonic operator is precise to degree  $d$ , given a point  $\mathbf{x}_0$ , when  $r \rightarrow +\infty$ ,  $\psi_j(r\mathbf{x}_0)$ , denoting the characteristic function in cell  $C_j$ , decays to zero at a rate of  $O(\frac{1}{r^d})$ .

*Proof.*

$$\begin{aligned} \psi_j(r\mathbf{x}_0) &= \sum_i n_{ij} \phi_{\mathbf{t}_i}(r\mathbf{x}_0) = r \sum_i n_{ij} \phi_{r^{-1}\mathbf{t}_i}(\mathbf{x}_0) \\ &= r \sum_i n_{ij} \phi_{\mathbf{x}_0}(r^{-1}\mathbf{t}_i), \end{aligned}$$



**Figure 3:** When summing the discrete bi-Laplacian operator, the inner edges (black arrows) vanish leaving just the boundary sums (blue arrows).

where  $n_{ij}$  is the coefficients of the biharmonic operator in cell  $C_j$ . Since  $\sum_i n_{ij} = 0$ , it can be derived that

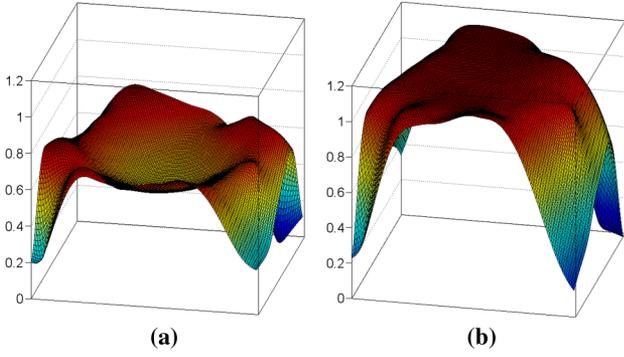
$$\lim_{r \rightarrow +\infty} r \sum_i n_{ij} \phi_{\mathbf{x}_0}(r^{-1}\mathbf{t}_i) = 0.$$

Therefore, if we expand  $\sum_i n_{ij} \phi_{\mathbf{x}_0}(r^{-1}\mathbf{t}_i)$  with variable  $r^{-1}$  at point 0 by Taylor expansion, it converges to 0 at a rate of  $O(\frac{1}{r^{d+1}})$  when  $r \rightarrow +\infty$ . So  $\psi_j(r\mathbf{x}_0)$  converges to 0 at a rate of  $O(\frac{1}{r^d})$ .  $\square$

The discrete biharmonic operator decays fast towards zero, approximating the characteristic function, which is localized.

### 5.2. Partition of Unity

The sum of all the basis functions approximates one and thus approximates a partition of unity as shown in Figure 2(b). As described in Section 4.1, the sum of all the characteristic functions equals to one for any fixed point  $\mathbf{y} \in \Omega$ , and otherwise it equals to zero for any fixed point  $\mathbf{y} \notin \Omega$ . Our discrete biharmonic basis functions approximate these characteristic functions. So the sum of all the basis functions should approximate 1 and thus approximate a partition of unity. Moreover, we give another interpretation from another point of view as follows. Since  $\psi_j(\mathbf{x}) = \mathbf{H}_{jj} \Delta \phi_{\mathbf{t}_j}(\mathbf{x}) + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_k}(\mathbf{x})$  and  $\mathbf{H}_{jj} + \sum_{\mathbf{t}_k \in \mathcal{N}_j} \mathbf{H}_{kj} = 0$ ,  $\psi_j(\mathbf{x}) = \sum_{\mathbf{t}_k \in \mathcal{N}_j} (\mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_k}(\mathbf{x}) - \mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_j}(\mathbf{x}))$ . In other words, it is the sum of differences between the central knot and all of its neighbours. Though the term  $\mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_k}(\mathbf{x}) - \mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_j}(\mathbf{x})$  is derived from the global optimization of Equation (16), it should also approximate the integral of the characteristic function on the common Voronoi face of adjacent knots  $\mathbf{t}_k$  and  $\mathbf{t}_j$ . The term  $\mathbf{H}_{jk} \Delta \phi_{\mathbf{t}_j}(\mathbf{x}) - \mathbf{H}_{jk} \Delta \phi_{\mathbf{t}_k}(\mathbf{x})$  is opposite to  $\mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_k}(\mathbf{x}) - \mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_j}(\mathbf{x})$  since  $\mathbf{H}_{kj} = \mathbf{H}_{jk}$ . So when summing all of the bases, the differences between inner adjacent knots (black arrows in Figure 3) vanish, such that  $\sum_i \psi_i(\mathbf{x}) = \sum_{\mathbf{e}_{kj} \in (\mathcal{T}' - \mathcal{T}, \mathcal{T})} (\mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_k}(\mathbf{x}) - \mathbf{H}_{kj} \Delta \phi_{\mathbf{t}_j}(\mathbf{x}))$ , which approximates the boundary integral of the region (blue arrows in Figure 3). Since the boundary integral of Equation (8) equals to one, the sum of basis functions approximates a partition of unity.



**Figure 4:** Illustration of the effect of the intrinsic constraint of the Laplacian operator. (a) Without the constraint. (b) With the constraint.

**5.3. Constraints**

The intrinsic property (i.e. the sum of coefficients of the Laplacian operator vanishes) of Laplacian operator is necessary for the optimization. We compare the results with and without the constraints in Equations (14) and (16) to illustrate its effects. Figure 4 shows the sum of bases computed by ignoring the constraints and the original form. It is clear that the sum of bases is unsmooth without the constraints compared to the original form due to overfitting to the object functions with fewer constraints.

**6. Qualitative and Quantitative Evaluation and Comparison of Basis Functions**

We evaluate and compare our method with respect to Feng’s method [FW12] which can be generalized to 3D domain directly. They compute the bi-Laplacian operator by iterated discrete Laplacian followed by refining it to cubic precision under cubic constraints. In Section 6.1, we compare the overall precision of our method to Feng’s method, and in Section 6.2, we compare the specific basis functions with different sampling knots.

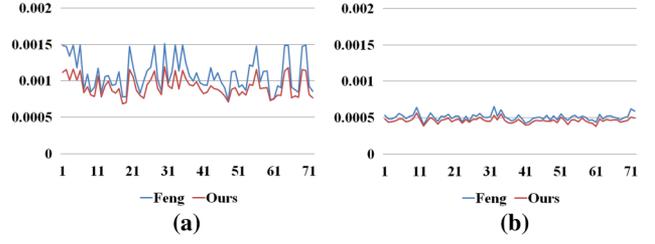
**6.1. Basis Function Precision**

The error of the discrete operator centred at knot  $\mathbf{t}_j$  at point  $\mathbf{x}$  is  $e = \psi_{\mathbf{t}_j}(\mathbf{x}) - \delta_{\mathbf{t}_j}(\mathbf{x})$ , where  $\delta_{\mathbf{t}_j}(\mathbf{x}) = 1$  if  $\mathbf{x}$  is in the Voronoi cell of  $\mathbf{t}_j$ , and otherwise  $\delta_{\mathbf{t}_j}(\mathbf{x}) = 0$ . It is a rational assumption that the errors have a Gaussian distribution  $N(0, \sigma^2)$  with mean 0 and variance  $\sigma^2$ . Given a set of fitting errors  $\{e_1, \dots, e_n\}$ , the joint probability density function is

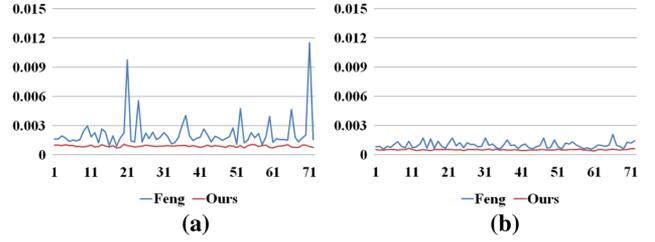
$$\mathcal{L}(X_1 = e_1, \dots, X_n = e_n) = \prod_{i=1}^n f_g(X = e_i) = \frac{1}{\sigma \sqrt{2\pi}} \prod_{i=1}^n e^{-\frac{e_i^2}{2\sigma^2}},$$

$$\ln \mathcal{L}(X_1 = e_1, \dots, X_n = e_n) = -\frac{1}{\sigma^2} \sum_{i=1}^n e_i^2 + \ln \frac{1}{\sigma \sqrt{2\pi}}. \tag{17}$$

It is evident that a larger joint probability density  $\mathcal{L}(X_1 = e_1, \dots, X_n = e_n)$  is equivalent to be with a smaller sum of squared errors  $\sum_{i=1}^n e_i^2$ . Thus, given two groups of fitting errors at different



**Figure 5:** Mean squared error comparison of our method and Feng’s algorithm. The horizontal axis is the region index and the vertical axis is the mean squared error.



**Figure 6:** Mean squared error comparison of our method and Feng’s algorithm. The horizontal axis is the region index and the vertical axis is the mean squared error.

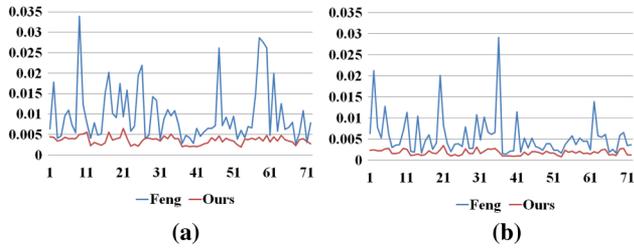
samples, we use the mean squared error to measure and evaluate their precisions. It also presents greater penalties to large errors compared to mean absolute errors.

For comparisons, we formulate the fitting errors as the following equation:

$$e = \frac{1}{|T| \times |\mathcal{S}|} \sum_{\mathbf{t}_i \in T} \sum_{\mathbf{x}_j \in \mathcal{S}} (\psi_{\mathbf{t}_i}(\mathbf{x}_j) - \delta_{\mathbf{t}_i}(\mathbf{x}_j))^2, \tag{18}$$

where the sample point set  $\mathcal{S} = \{\mathbf{x}_j | \mathbf{x}_j = \frac{1}{|\mathcal{N}_j|} \sum_{\mathbf{t}_k \in \mathcal{N}_j} (\mathbf{t}_j + \mathbf{t}_k)/2\}$ , i.e. we sample one point in every Voronoi cell, which is the average of the middle points of the centre and its neighbouring knots.

Given 72 regions, we construct biharmonic B-spline basis functions independently in every region to compare the basis errors. In this paper, all the regions are dilated by 3 voxels to satisfy the needs for intensity fitting as explained in Section 7 as well as filling small gaps near the boundaries. Figures 5–7 illustrate the comparisons to Feng’s method, in which the horizontal axis denotes the 72 regions (indices start at 0) and the vertical axis denotes the squared error computed by Equation (18). First, we sample the knots evenly with small random perturbations and the errors are shown in Figure 5(a) with about 40–90 bases in every region. The averages of errors are 0.001094 and 0.000923 using Feng’s and our methods, respectively. In Figure 5(b), we construct about 130–180 bases for every region and the averages of errors are 0.000509 and 0.000461, respectively. Secondly, we sample more knots near the original boundaries before dilation and the errors are shown in Figure 6(a) with about 140–180 bases in every region. The averages are 0.002207 and 0.000892,



**Figure 7:** Mean squared error comparison of our method and Feng's algorithm. The horizontal axis is the region index and the vertical axis is the mean squared error.

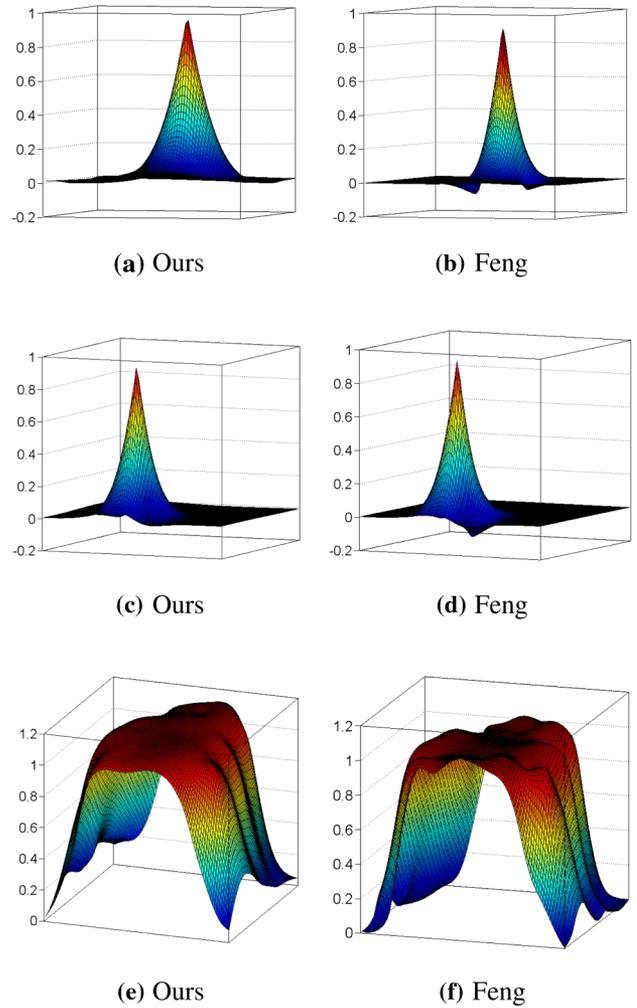
respectively. In Figure 6(b), we construct about 210–280 bases for every region and the averages are 0.00099 and 0.000501, respectively. Thirdly, we sample some knots (five) randomly and then sample other knots under (coordinate-wise) Gaussian distribution centred at the five knots, respectively, and the errors are shown in Figure 7(a) with about 20–80 bases in every region. The averages are 0.009882 and 0.003621, respectively. In Figure 7(b), we construct about 60–160 bases for every region and the averages are 0.005563 and 0.001791, respectively. In the first case, the two methods produce similar results and ours is a little better than Feng's. In the second and third cases, Feng's method becomes rather unstable as the result is fluctuating significantly. In contrast, our method in such cases is as good as the first case. It is obvious that our method produces more precise results than Feng's method, especially in the case of unevenly sampled knots, and our method is more robust to combat unevenly distributed knots.

**6.2. Basis Functions Comparison**

For a region with free boundary, Figures 8, 9 and 10 illustrate the basis functions and sum of basis functions on some cross-section planes in the region bounding box, under different sample knots, computed by our algorithm and Feng's algorithm, respectively. Similar to the last subsection, in Figure 8, the knots are evenly distributed with small random perturbation, we produce smoother basis functions as shown in Figures 8(a) and (c) compared to (b) and (d), which suffer from poles around the bases resulting from the uneven distributions of neighbouring knots. Our sum of bases appears smoother than Feng's result as shown in Figures 8(e) and 8(f).

In Figure 9, after evenly sampling, we sample more knots near the original boundary before dilation. In contrast to Figures 9(a) and 9(c), the unevenly distributed neighbours result in poles with extremely negative values using Feng's method as shown in Figures 9(b) and 9(d), respectively, the reason for which will be discussed later. The sum of bases are shown in Figures 9(e) and 9(f) respectively, where our result appears smoother than Feng's result.

In Figure 10, some knots (five) are sampled randomly and then other knots are sampled under (coordinate-wise) Gaussian distribution centred at the five knots, respectively, such that the knots distribute rather unevenly. We compute their basis functions as shown in Figures 10(a) and 10(c), respectively. Nevertheless, the maximum

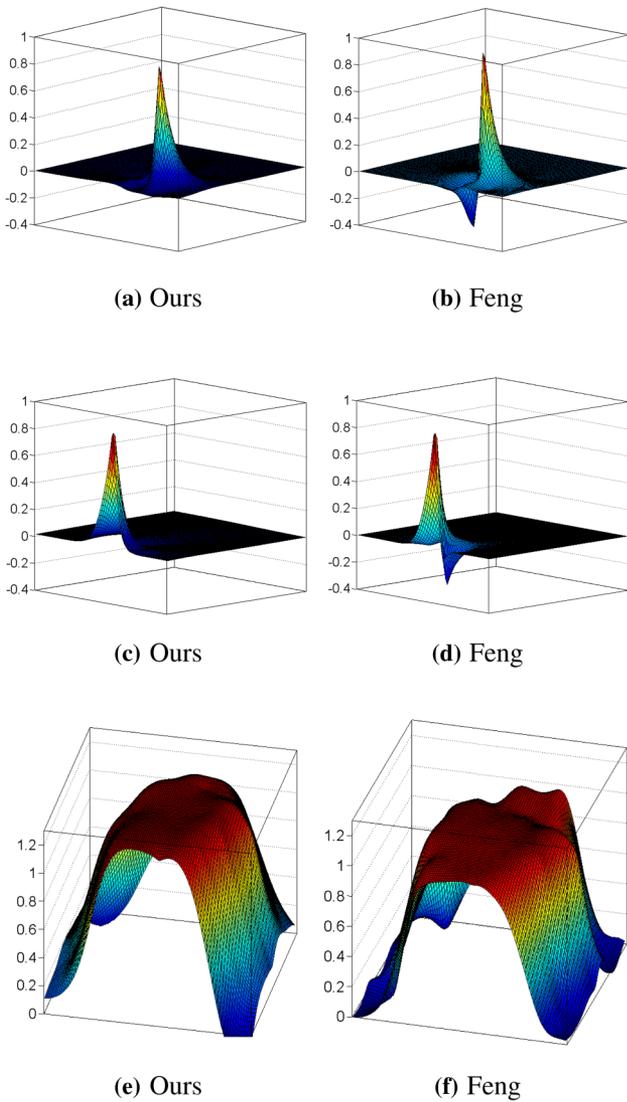


**Figure 8:** Basis functions and the sum of basis functions with evenly distributed knots.

has exceeded 1.5 and the pole with extremely negative appears in 10(b) using Feng's method and it also exceeds 1.4 in (d). Our sum of bases (Figure 10e) appears much smoother than Feng's result (Figure 10f) which has been nearly failing in case of extremely unevenly sampled knots.

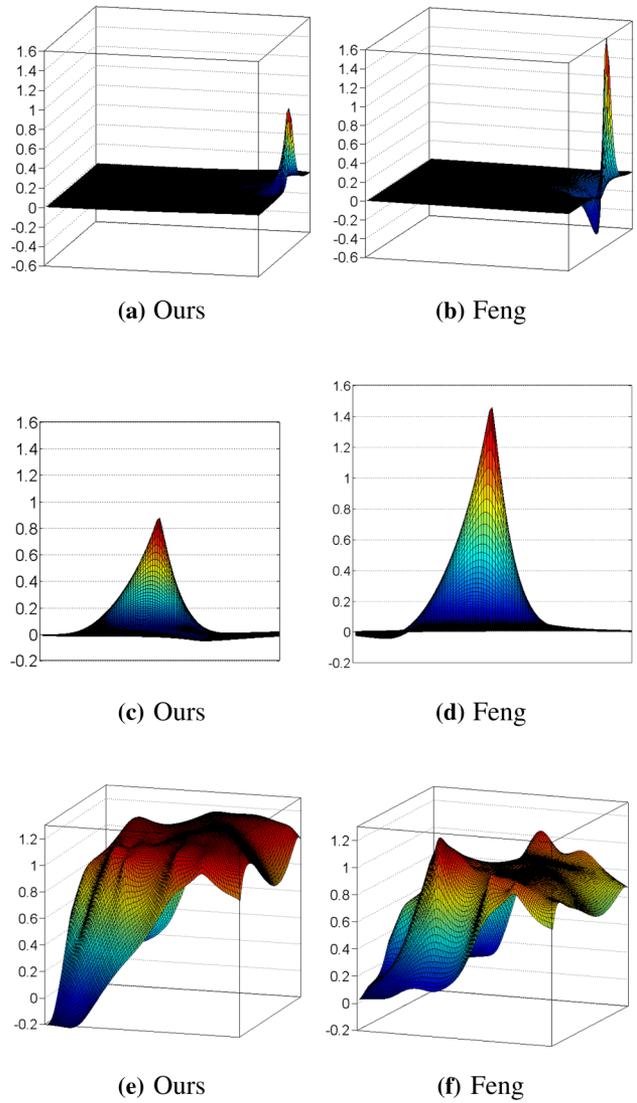
We observe that, when a neighbouring knot is relatively much too close to the central knot compared to other neighbouring knots, Feng's method would suffer from dramatic distortions, failing to construct the local basis. Hence, a pair of adjacent knots (that are too close) results in a pair of seriously distorted basis functions centred at the two knots. Figures 11(a) and 12(a) illustrate two basis functions evaluated by Feng's method centred at two very close knots as an example. The discrete biharmonic operator at knot  $\mathbf{t}_j$  is formulated as

$$\psi_j(\mathbf{x}) = -\frac{1}{8\pi} \sum_i n_{ij} \|\mathbf{x} - \mathbf{t}_i\|. \tag{19}$$

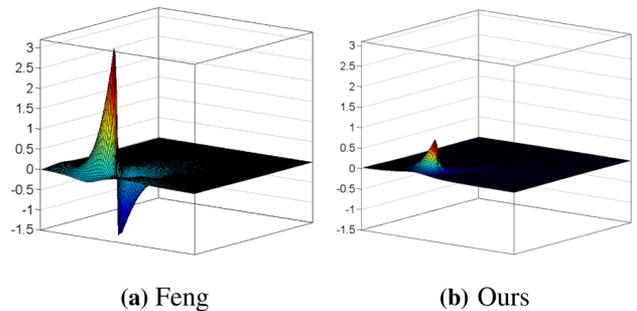


**Figure 9:** Basis functions and the sum of basis functions with a bit more knots randomly sampled in the interior of the region.

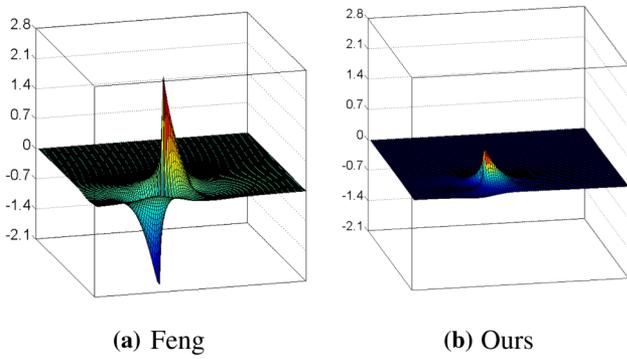
The summation of coefficients  $\sum_i n_{ij} = 0$ . Ideally, the coefficient of the central knot is a relatively large positive number and the others are relatively small around 0. However, because the distance between the two knots is the denominator of the coefficients (Equation 11), the unevenly distributed neighbouring knots result in biased negative coefficient (with a larger absolute value) for the neighbouring knot (that is too close) as well as very large positive coefficient for the central knot. It means the summation  $-\frac{1}{8\pi} \sum_i n_{ij} \|\mathbf{x} - \mathbf{t}_j\|$  is dominated by the two terms, which gives rise to extremely negative value (i.e. poles) around the neighbouring knots (that are too close) and extremely large value (high peak) around the central knot. We evaluate the values of the basis functions at these sample knots. In the first basis, the values reach 3.61 at the central knot and  $-1.89$  at the neighbouring knots. Analogously, in the second basis function,



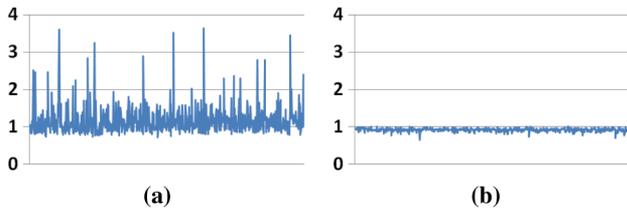
**Figure 10:** Basis function and the sum of basis functions with knots randomly sampled around some centres.



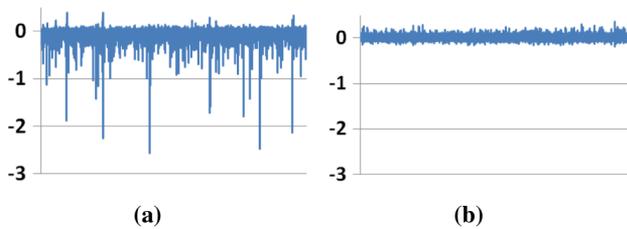
**Figure 11:** (a) The extremely distorted basis by Feng's method. (b) The smooth basis by our method.



**Figure 12:** (a) The extremely distorted basis by Feng's method. (b) The smooth basis by our method.



**Figure 13:** The values of the basis functions at the central knots. Vertical axis is the function value and the horizontal axis is the central knots' indices. (a) Feng's method. (b) Our method.



**Figure 14:** The value distribution of the basis functions at the neighbouring knots in the region. Vertical axis is the function value and the horizontal axis is the neighbouring knots' indices. (a) Feng's method. (b) Our method.

the values are 2.90 at the central knot and  $-2.57$  at the neighbouring knot. The two figures illustrate the basis functions on a plane crossing the central knots rather than in the entire 3D space, so the extremal values are different from the above values (due to the discrete sampling, the central knots may not happen to be sampled).

In contrast, our optimization method distributes the error more evenly and produces much smoother basis functions with slight distortion as shown in Figures 11(b) and 12(b). The values of basis are 0.931 and 0.111 at the central and neighbouring knots, respectively, in the first basis function. Meanwhile, they are 0.940 and  $-0.125$ , respectively, in the second one.

Figures 13 and 14 illustrate the values of all the basis functions at the central and neighbouring knots (excluding ghost knots),

respectively, where the horizontal axes denote the knot indices and the vertical axes denote the values. The values at the centroid knots should be around 1. The average difference to 1 is 0.0799 using our method, but instead, it is 0.2481 using Feng's method. On the other hand, the values at the neighbouring knots should be around 0. The average difference to 0 is 0.01932 using our method, but it is 0.02167 using Feng's method. Many of the values of Feng's method are moving out of the reasonable range. It clearly shows that our method is more stable and generates much better results than Feng's method.

## 7. Application and Comparison

We use the biharmonic B-splines to fit 3D volume data to demonstrate the usefulness of these basis functions, and then make comparison to Feng's algorithm [FW12]. The functions represented by the basis functions are continuous. In order to preserve the discontinuity of the volume intensities, we segment the volume into supervoxels (regions) [ASS\*12] and then fit the volume intensities independently.

### 7.1. Supervoxel Intensity Fitting

After segmentation into supervoxels, the intensities of every supervoxel are fitted independently. The intensity function of every supervoxel  $\mathcal{V}_i$  is formulated as  $\Phi^i(\mathbf{x})\mathbf{H}^i\mathbf{m}^i$ , where  $\Phi^i(\mathbf{x}) = [\phi_{\mathbf{t}_1}(\mathbf{x}), \phi_{\mathbf{t}_2}(\mathbf{x}), \dots, \phi_{\mathbf{t}_{|T^i|}}(\mathbf{x})]$ ,  $\mathbf{m}^i$  is the coefficient vector of the  $i$ th supervoxel and  $\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_{|T^i|}$  are the  $|T^i|$  sample knots (including ghost knots) of the supervoxel  $\mathcal{V}_i$ . Since each basis function is localized, it is adequate to approximate the supervoxel intensities. For each supervoxel  $\mathcal{V}_i$ , we fit its intensities by minimizing the following energy function:

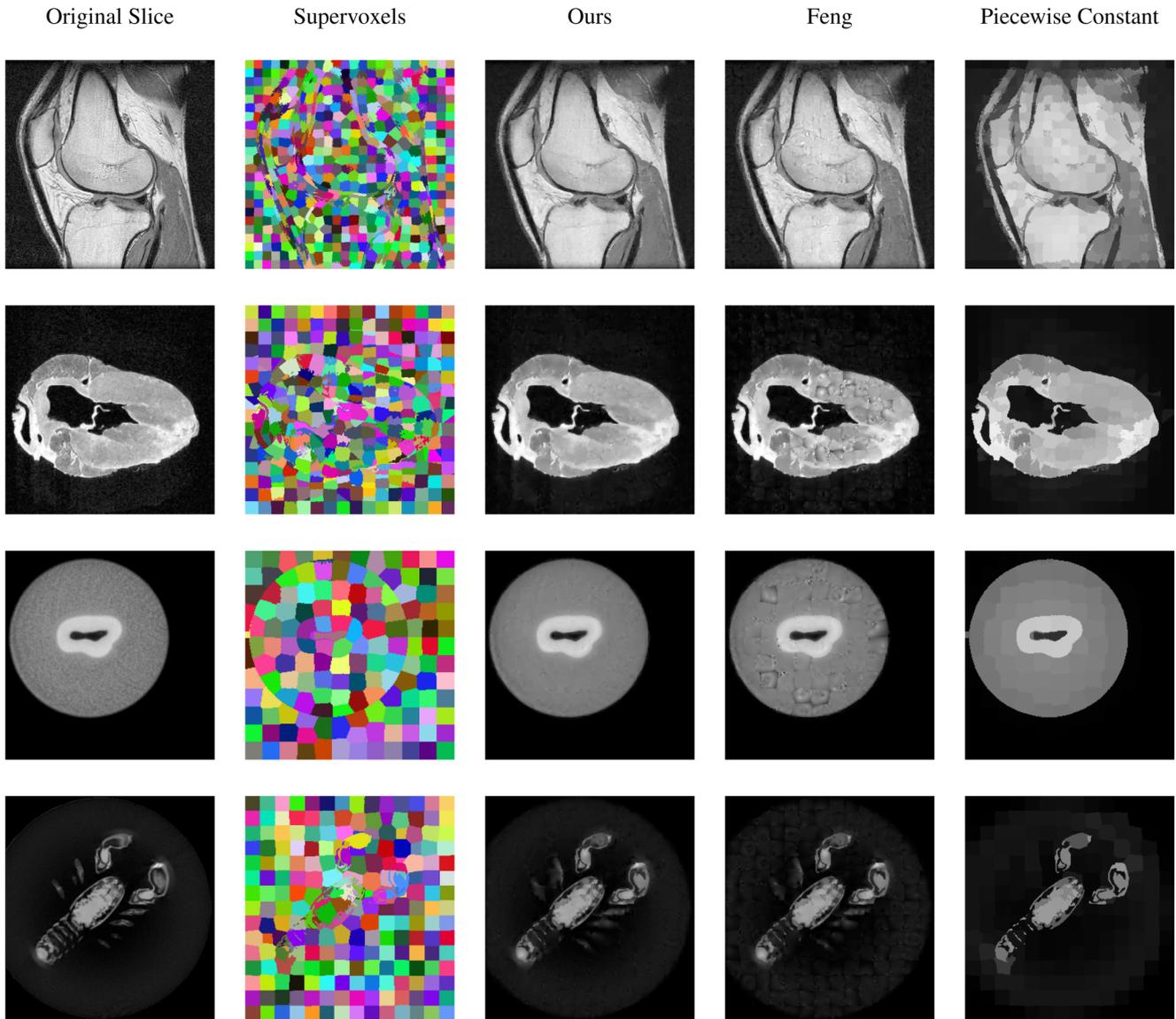
$$\mathbf{m}^i = \arg \min_{\mathbf{m}^i} \sum_{\mathbf{x}_j \in \mathcal{V}_i} (\Phi^i(\mathbf{x}_j)\mathbf{H}^i\mathbf{m}^i - c(\mathbf{x}_j))^2, \quad (20)$$

where  $c(\mathbf{x}_j)$  is the intensity of the point  $\mathbf{x}_j$ .

### 7.2. Fitting Results

We use synthetic and real volume images to test and compare our algorithm to Feng's [FW12]. We use two synthetic volume images, monochromatic and gradient volume images, for our experiments at the beginning.

We synthesize a  $128 \times 128 \times 80$  monochromatic volume image with intensity 200. The results are documented in Table 1, where four different numbers of basis functions are sampled for comparison. The number of basis functions divided by the number of voxels is documented in the first row and the second and third rows document the relative error of our method and Feng's algorithm, respectively. The relative error is defined as the average of all the fitted error divided by the real intensity 200. It is formulated as  $(\frac{1}{n} \sum_i |v_i - 200|)/200$ , where  $n$  is the number of voxels and  $v_i$  is the fitted value of the  $i$ th voxel. It is apparent that our algorithm produces much more precise results compared to Feng's algorithm.



**Figure 15:** Comparisons of some of the fitted slices using the bases constructed by our algorithm and Feng’s algorithm as well as piecewise constant intensity, respectively. Every supervoxel is coloured randomly. The volume data are Knee (row 1), Heart (row 2), Tooth (row 3), Lobster (row 4), and the data sets courtesy of <http://www.stereofx.org/>.

**Table 1:** Statistics of fitting of monochromatic volume image.

	Case1	Case2	Case3	Case4
Bases/Voxels	4.79%	3.05%	1.88%	1.23%
Ours	0.367%	0.904%	1.34%	2.27%
Feng	0.524%	1.17%	1.85%	3.19%

**Table 2:** Statistics of fitting of gradient volume image.

	Case1	Case2	Case3	Case4
Bases/Voxels	4.65%	3.06%	1.88%	1.21%
Ours	0.249%	0.461%	0.692%	1.25%
Feng	0.323%	0.589%	0.943%	1.67%

Analogously, we synthesize a gradient volume image and the results are documented in Table 2. The relative error is defined as  $(\frac{1}{n} \sum_i^n |v_i - v'_i|)/255$ , where the  $v_i$  and  $v'_i$  denote the fitted intensity

and original intensity of the  $i$ th voxel, respectively, and 255 is the maximum intensity. Also, it is obvious that our algorithm produces much more precise results compared to Feng’s algorithm.

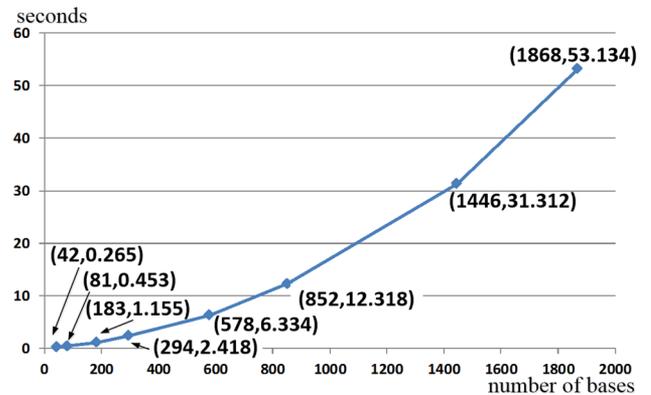
**Table 3:** Statistics of real volume data.

Model	Resolution	Supervoxels	Bases (Bases/Voxels)	Knots	Error (Ours)	Error (Feng)	Error (Constant)
Knee	512×512×87	2374	499 136 (2.19%)	2 042 723	4.56%	4.89%	6.00%
Heart	352×352×256	3057	464 679 (1.46%)	2 210 665	1.91%	2.23%	2.56%
Tooth	256×256×161	1008	228 279 (2.16%)	1 022 267	1.12%	1.52%	1.75%
Lobster	301×324×56	420	64 193 (1.18%)	330 513	0.844%	1.019%	1.747%

We also test four real volume data and make comprehensive comparisons. Some slices of the four real volume data and their fitted slices are illustrated in Figure 15 (Please refer to the attached video for more slices). The volumes are segmented into supervoxels as shown in the second column and they are fitted independently to preserve intensity boundaries (i.e. discontinuity). For comparison, we show the results by assigning a constant intensity (i.e. the average intensity of the supervoxel) for every supervoxel. In every supervoxel, we sample five points randomly and sample the knots under (coordinate-wise) Gaussian distribution centred at the five points randomly, giving rise to highly unevenly distributed knots. In order to hold enough support for supervoxel's boundary, we extend the boundary outward for 3 voxels when sampling knots, resulting in some basis centres outside the supervoxel's boundaries within 3 voxels intentionally. The basis functions computed by our method exhibit much better fitting results than Feng's algorithm especially on the supervoxels' boundaries, as illustrated in the third and fourth columns of Figure 15. The fifth column shows the piecewise constant fitting slices (i.e. every supervoxel is filled by its average intensity) to illustrate the contributions of supervoxels. The statistics are documented in Table 3. The error is evaluated similar to the case of the gradient volume image. It is obvious that our basis functions produce much more precise fitted volumes compared to Feng's algorithm and piecewise constant fitting method.

It may be noted that, the thin-plate spline is also formulated as a linear combination of Green's functions capable of interpolating/fitting a data set for graphics applications. It interpolates the function values (e.g. volume intensities) while minimizing the bending energy according to the values at the sparsely sampled knots, but, it is inappropriate to fit the volume intensities without considering the values at other voxels. Instead, we fit the volume intensities using basis functions constructed by Green's functions according to Equation (20).

**Performance.** We use Gurobi (<http://www.gurobi.com>) as the quadratic programming solver. Our algorithm is implemented in C++, which runs on an Intel i7-3770 CPU and partially on an NVIDIA GTX 680 GPU to assemble matrices. The timing statistics for different numbers of bases are documented in Figure 16. Because of the global optimization in Equation (16), the running time increases a bit faster than linear function as the numbers of bases increase. It may be noted that, we have not yet made extensive efforts to optimize the programme/code and these documented data also include timing to perform several data swapping operations. The Feng's algorithm for comparison is implemented in Matlab and uses Qhull (<http://www.qhull.org>) to compute the Voronoi diagram.

**Figure 16:** The time performance of our algorithm.

**Limitation.** There still remain some problems necessary for further improvement. The quadratic programming in Equation (16) is a global optimization involving a large number of variables, and in principle it prevents the number of knots from increasing to a large value. Another issue is that, the maximum of sum of basis functions in our method may reach to 1.2 to 1.3, which essentially is a compromise to the smoothness of the basis functions.

## 8. Conclusion and Future Work

In this paper, we have articulated a novel method to formulate the discrete bi-Laplacian operator as a quadratic programming problem optimized for a specific function set, which is more suitable for 3D bi-Laplacian than the existing 2D methods. Our algorithm is especially robust and stable when the knots are sampled unevenly than Feng's algorithm [FW12], which has been demonstrated extensively throughout this paper using various examples. We have also applied our novel trivariate biharmonic B-splines to volume image fitting and compared our method to Feng's formulation based on our comprehensive experiments in both synthetic and real volume images, which have demonstrated the superiority of our new formulation. The unevenly sampled knots may be useful for adaptive fitting with different accuracies on different parts of the volume. Except for the intensity fitting, the biharmonic B-spline may also be used for supervoxel shape fitting, volume data simplification, etc.

As for our ongoing and near-future research efforts, we are planning to implement our newly designed algorithms completely on CUDA to continue to improve the computational efficiency to the maximum extent. And we also plan to generalize the biharmonic

B-splines to higher dimensional space and explore more applications using the biharmonic B-splines to extend their application scopes.

### Acknowledgements

The authors would like to thank the reviewers for their helpful comments. This work is supported partially by National Natural Science Foundation of China (Grant No. 61300068, 61190120, 61190121, 61190125, 61300067), National Science Foundation of the United States (Grant No. IIS-0949467, IIS-1047715, and IIS-1049448), the National High Technology Research and Development Programme (863 Program) of China (Grant No. 012AA011503) and Postdoctoral Science Foundation of China (Grant No. 2013M530512).

### References

- [ASS\*12] ACHANTA R., SHAJI A., SMITH K., LUCCHI A., FUA P., SÜSTRUNK S.: Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 11 (2012), 2274–2282.
- [Boo89] BOOKSTEIN F. L.: Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11, 6 (June 1989), 567–585.
- [Buh03] BUHMANN M. D.: *Radial Basis Functions: Theory and Implementations*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, 2003.
- [Chu97] CHUNG F. R.: *Spectral Graph Theory*, vol. 92. American Mathematical Society, 1997.
- [EG04] ERN A., GUERMOND J.-L.: Finite element interpolation. In *Theory and Practice of Finite Elements*, vol. 159 of Applied Mathematical Sciences. Springer New York, 2004, pp. 3–80.
- [FSH11] FINCH M., SNYDER J., HOPPE H.: Freeform vector graphics with controlled thin-plate splines. *ACM Transactions on Graphics* 30, 6 (Dec. 2011), 166:1–166:10.
- [FW12] FENG P., WARREN J.: Discrete bi-Laplacians and biharmonic B-splines. *ACM Transactions on Graphics* 31, 4 (July 2012), 115:1–115:11.
- [HP11] HILDEBRANDT K., POLTHIER K.: On approximation of the Laplace-Beltrami operator and the Willmore energy of surfaces. *Computer Graphics Forum* 30, 5 (2011), 1513–1520.
- [JMD\*07] JOSHI P., MEYER M., DEROSE T., GREEN B., SANOCKI T.: Harmonic coordinates for character articulation. *ACM Transactions on Graphics* 26, 3 (July 2007) Article No. 71.
- [JSW05] JU T., SCHAEFER S., WARREN J.: Mean value coordinates for closed triangular meshes. *ACM Transactions on Graphics* 24, 3 (July 2005), 561–566.
- [JTSZ10] JACOBSON A., TOSUN E., SORKINE O., ZORIN D.: Mixed finite elements for variational surface modeling. *Computer Graphics Forum* 29, 5 (2010), 1565–1574.
- [Lev98] LEVIN D.: The approximation power of moving least-squares. *Mathematics of Computation* 67, 224 (Oct. 1998), 1517–1531.
- [LJH13] LI X.-Y., JU T., HU S.-M.: Cubic mean value coordinates. *ACM Transactions on Graphics* 32, 4 (July 2013), 126:1–126:10.
- [LRF10] LIPMAN Y., RUSTAMOV R. M., FUNKHOUSER T. A.: Biharmonic distance. *ACM Transactions on Graphics* 29, 3 (July 2010), 27:1–27:11.
- [MDSB03] MEYER M., DESBRUN M., SCHRÖDER P., BARR A.: Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III*, Mathematics and Visualization. Springer Berlin Heidelberg, 2003, pp. 35–57.
- [ME10] MIRZARGAR M., ENTEZARI A.: Voronoi splines. *IEEE Transactions on Signal Processing* 58, 9 (2010), 4572–4582.
- [OBA\*03] OHTAKE Y., BELYAEV A., ALEXA M., TURK G., SEIDEL H.-P.: Multi-level partition of unity implicits. *ACM Transactions on Graphics* 22, 3 (July 2003), 463–470.
- [Ryc09] RYCROFT C. H.: Voro++: A three-dimensional Voronoi cell library in C++. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 19, 4 (2009), 041111.
- [SOG09] SUN J., OVSIANIKOV M., GUIBAS L.: A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum* 28, 5 (2009), 1383–1392.
- [SZBN03] SEDERBERG T. W., ZHENG J., BAKENOV A., NASRI A.: T-splines and T-NURCCs. *ACM Transactions on Graphics* 22, 3 (July 2003), 477–484.
- [Wen05] WENDLAND H.: *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2005.
- [WHL\*08] WANG H., HE Y., LI X., GU X., QIN H.: Polycube splines. *Computer-Aided Design* 40, 6 (June 2008), 721–733.
- [WMKG07] WARDETZKY M., MATHUR S., KÄLBERER F., GRINSPUN E.: Discrete Laplace operators: No free lunch. In *Proceedings of the Symposium on Geometry Processing (Aire-la-Ville, Switzerland, Switzerland, 2007)*, Eurographics Association, pp. 33–37.
- [WPG12] WEBER O., PORANNE R., GOTSCHMAN C.: Biharmonic coordinates. *Computer Graphics Forum* 31, 8 (Dec. 2012), 2409–2422.

### Supporting Information

Additional Supporting Information may be found in the online version of this article at the publisher's web site:

### Video S1