Convergent Discrete Laplace-Beltrami Operators over Triangular Surfaces

Guoliang Xu *
Academy of Mathematics and System Sciences,
Chinese Academy of Sciences, Beijing, China
Email: xuguo@lsec.cc.ac.cn

Abstract

The convergence property of the discrete Laplace-Beltrami operators is the foundation of convergence analysis of the numerical simulation process of some geometric partial differential equations which involve the operator. In this paper we propose several simple discretization schemes of Laplace-Beltrami operators over triangulated surfaces. Convergence results for these discrete Laplace-Beltrami operators are established under various conditions. Numerical results that support the theoretical analysis are given. Application examples of the proposed discrete Laplace-Beltrami operators in surface processing and modelling are also presented.

Key words: Laplace-Beltrami Operator; Surface triangulation; Discretization; Convergence.

1 Introduction

Laplace-Beltrami operator, abbreviated as LBO in this paper, is a generalization of the Laplacian from flat spaces to manifolds. LBO plays a central role in many areas, such as image processing (see [4, 13, 20, 27]), signal processing (see [25, 26]), surface processing (see [2, 7, 8, 9, 21, 22]), and the study of geometric partial differential equations (PDE) (see [18, 4, 15, 20]). For instance, the mathematical formulation of the mean curvature flow, surface diffusion flow (see [15]) and Willmore flow (see [23]) etc. involves the first and second order LBOs. In solving numerically PDE which involves the classical Laplacian on flat spaces, a standard technique is to approximate the Laplacian by a finite divided difference. Likewise, the LBO needs to be discretized in solving the geometric PDEs numerically on surfaces. However, due to the complexity and the diversity of the discretized surfaces, the discretization of the LBO is not as simple as the Laplacian over the flat space. In the literature, several discretizations of LBO over surfaces have been proposed and used. However, to the best of author's knowledge, none of these discretizations has been proved to be convergent as the size of surface mesh goes to zero.

The convergence of the discrete LBOs is the foundation for the convergence of some numerical simulation process of PDE which involves the LBO. In this paper we propose several discretization schemes of the LBOs over triangulated surfaces. Convergence results for these discrete LBOs are obtained under various special conditions. We also review several already used discrete LB operators including Taubin's discretization (see [25], 1995; [26], 2000), Fujiwara's discretization (see [10], 1995), Desbrun et al's discretization (see [8], 1999), Mayer's discretization (see [15], 2001), Meyer et al's discretization (see [16], 2002), and Desbrun et al's discretization (see [9], 2000).

It is well known that LB operator relates closely to the mean curvature normal (see (2.6)). Hence, an approximation of mean curvature normal may lead to a discretization of the LBO. On the approximation of curvatures, there exist also many approaches, such as the ones proposed by Chen, Hamann and Taubin to name a few [6, 12, 24]. However, these approaches do not yield the linear form as (2.7).

The remaining of the paper is organized as follows. In Section 2, we introduce some basic material on LBO and then review several existing discretizations of the operator. In Section 3, we propose

*Support in part by NSFC grants 10241004, 10371130, National Innovation Fund 1770900, Chinese Academy of Sciences.

several alternatives of the discretization and establish some convergence results. Numerical examples for comparing these discrete operators are given in Section 5. Possible applications of these discrete operators are described in Section 6. Section 7 concludes the paper. Some proofs of the theoretical results are put into Appendix.

2 LBO and its Discretization

To describe the Laplace-Beltrami operator over surfaces precisely, let us introduce some terminology and notations. Let $\mathcal{M} \subset I\!R^3$ be a two-dimensional manifold, and $\{U_\alpha, x_\alpha\}$ be the differentiable structure. The mapping x_α with $x \in x_\alpha(U_\alpha)$ is called a parameterization of $\mathcal{M}$ at x. Denoting the coordinate U_α as (ξ_1, ξ_2), then the tangent space $T_x\mathcal{M}$ at $x \in \mathcal{M}$ is spanned by $\{\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2}\}$. For a given point $x \in x_\alpha(U_\alpha) \subset \mathcal{M}$, the tangent vector components $\frac{\partial}{\partial \xi_1}$ and $\frac{\partial}{\partial \xi_2}$ depend upon α, but $T_x\mathcal{M}$ does not. The set $T\mathcal{M} = \{(x, v);\ x \in \mathcal{M},\ v \in T_x\mathcal{M}\}$ is called a tangent bundle. Let $f \in C^2(\mathcal{M})$. The Laplace-Beltrami operator $\Delta_\mathcal{M}$ applying to f is defined by the duality

$$(\Delta_\mathcal{M} f, \phi)_\mathcal{M} = -(\nabla_\mathcal{M} f, \nabla_\mathcal{M} \phi)_{T\mathcal{M}} \tag{2.1}$$

for all $\phi \in C^\infty(\mathcal{M})$, where $\nabla_\mathcal{M}$ is the gradient operator, which is given by (see [5], page 102)

$$\nabla_\mathcal{M} f = [\ t_1,\ t_2\ ] G^{-1} \nabla f \in I\!R^3, \tag{2.2}$$

where $\nabla f = \left[\ \frac{\partial f(x(\xi_1,\xi_2))}{\partial \xi_1},\ \frac{\partial f(x(\xi_1,\xi_2))}{\partial \xi_2}\ \right]^T \in I\!R^2$, $G = [t_1,\ t_2]^T[t_1,\ t_2] = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}$, $g_{ij} = \langle t_i, t_j \rangle$ and $t_i = \frac{\partial x}{\partial \xi_i}$ are the tangent vectors. The gradient $\nabla_\mathcal{M} f$ is geometric intrinsic, though the expression (2.2) depends on a local surface parameterization. That is we have the following lemma:

Lemma 2.1 *Let $S \in I\!R^{2\times 2}$ be a nonsingular matrix, and*

$$[\tilde{t}_1, \tilde{t}_2] = [t_1,\ t_2]S, \quad \tilde{\nabla} f = \nabla f\ S.$$

Then $[\tilde{t}_1, \tilde{t}_2]\tilde{G}^{-1}\tilde{\nabla} f = \nabla_\mathcal{M} f$, where $\tilde{G} = [\tilde{t}_1, \tilde{t}_2]^T[\tilde{t}_1, \tilde{t}_2]$.

The inner products in (2.1) are given by

$$\begin{aligned} (f, g)_\mathcal{M} &= \int_\mathcal{M} fg dx, \quad f, g \in C^0(\mathcal{M}), \\ (\phi, \psi)_{T\mathcal{M}} &= \int_\mathcal{M} \langle \phi, \psi \rangle dx, \quad \phi, \psi \in T\mathcal{M}. \end{aligned}$$

A simple computation leads to the following representations of $\Delta_\mathcal{M} f$:

$$\begin{aligned} \Delta_\mathcal{M} f &= \frac{1}{\sqrt{g}} \sum_{ij} \frac{\partial}{\partial \xi_i} \left(\sqrt{g}\ g^{ij} \frac{\partial f}{\partial \xi_j} \right) \\ &= \frac{1}{2g} \left[\frac{\partial g}{\partial \xi_1}, \frac{\partial g}{\partial \xi_2} \right] G^{-1} \nabla f + \left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2} \right] \left(G^{-1} \nabla f \right) \qquad (2.3) \\ &= \frac{1}{2g} \left[\frac{\partial g}{\partial \xi_1}, \frac{\partial g}{\partial \xi_2} \right] G^{-1} \nabla f + \left(\left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2} \right] G^{-1} \right) \nabla f \\ &\quad + g^{11} \frac{\partial^2 f}{\partial \xi_1^2} + 2g^{12} \frac{\partial^2 f}{\partial \xi_1 \partial \xi_2} + g^{22} \frac{\partial^2 f}{\partial \xi_2^2}, \qquad (2.4) \end{aligned}$$

where g^{ij} is defined by $G^{-1} = (g^{ij})_{ij}$ and $g = \det(G)$. Let $t_{ij} = \frac{\partial^2 x}{\partial \xi_i \partial \xi_j}$, $g_{ijk} = \langle t_i, t_{jk} \rangle$. Then (2.3) could be written as

$$\Delta_\mathcal{M} f = \frac{1}{g} \left[\begin{array}{c} g_{11}g_{212} + g_{22}g_{111} - g_{12}(g_{211} + g_{112}) \\ g_{11}g_{222} + g_{22}g_{112} - g_{12}(g_{212} + g_{122}) \end{array} \right]^T G^{-1} \nabla f + \left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2} \right] \left(G^{-1} \nabla f \right). \tag{2.5}$$

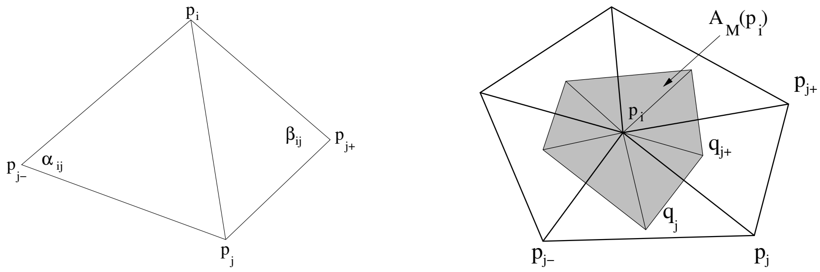


Fig 2.1: Left: The definition of the angles α_{ij} and β_{ij}. Right: The definition of the area $A_M(p_i)$.

Let $\text{div}_{\mathcal{M}}\psi$ denote the divergence for a vector field $\psi \in T\mathcal{M}$, which is defined as the dual operator of the gradient (see [19]):

$$(\text{div}_{\mathcal{M}} v, \phi)_{\mathcal{M}} = -(v, \nabla_{\mathcal{M}}\phi)_{T\mathcal{M}}, \quad \forall \phi \in C_0^{\infty}(\mathcal{M}),$$

where $C_0^{\infty}(\mathcal{M})$ is a subspace of $C^{\infty}(\mathcal{M})$, whose elements have compact support. Then it is easy to see that $\text{div}_{\mathcal{M}}\nabla_{\mathcal{M}} = \Delta_{\mathcal{M}}$. Let p be a surface point of $\mathcal{M}$. Then it is known that (see [28], page 151)

$$\Delta_{\mathcal{M}} p = 2H(p) \in I\!R^3, \tag{2.6}$$

where $H(p)$ is the mean curvature normal at p. i.e., $\|H(p)\|$ is the mean curvature, $H(p)/\|H(p)\|$ is the unit surface normal.

Now we consider the discretization of $\Delta_{\mathcal{M}} p$. Let M be a triangulation of surface $\mathcal{M}$. Let $\{p_i\}_{i=1}^N$ be the vertex set of M. For vertex p_i with valence n, denote by $N_1(i) = \{i_1, i_2, \cdots, i_n\}$ the set of the vertex indices of one-ring neighbors of p_i. We assume in the following that these $i_1, \cdots, i_n$ are arranged such that the triangles $[p_i p_{i_k} p_{i_{k-1}}]$ and $[p_i p_{i_k} p_{i_{k+1}}]$ are in M, and p_{i_k}, $p_{i_{k+1}}$ opposite to the edge $[p_i p_{i_k}]$. For $j = i_k \in N_1(i)$, we use j_+ and j_- to denote i_{k+1} and i_{k-1}, respectively, for simplifying the notation. Furthermore, we use the following convention:

$$i_{n+1} = i_1, \quad i_0 = i_n.$$

Now we review several existing discretizations of LBO over triangular surfaces.

1. Taubin et al's Discretization (see [25], 1995; [8], 1999; [26], 2000; [17], 2002).
This is a class of discretizations in the following form

$$\Delta_M^{(1)} f(p_i) = \sum_{j \in N_1(i)} w_{ij}(f(p_j) - f(p_i)), \tag{2.7}$$

where the weights w_{ij} are positive numbers and $\sum_{j \in N_1(i)} w_{ij} = 1$. There are several ways to determine the weights. A simple way is to take $w_{ij} = 1/|N_1(i)|$, where $|\cdot|$ denotes the cardinality of a set. A more general way is to define them by a positive function ϕ: $w_{ij} = \phi(p_i, p_j)/\sum_{k \in N_1(i)} \phi(p_i, p_k)$, and function $\phi(p_i, p_j)$ can be the surface area of the two faces that share the edge $[p_i p_j]$, or some power of the length of the edge: $\phi(p_i, p_j) = \|p_i - p_j\|^{\alpha}$. Fujiwara take $\alpha = -1$ (see [10]). Desbrun et al's (see [8], 1999) define w_{ij} as $w_{ij} = \cot\alpha_{ij} + \cot\beta_{ij}/\sum_{k \in N_1(i)} \cot\alpha_{ik} + \cot\beta_{ik}$, where α_{ij} and β_{ij} are the triangle angles as shown in Fig 2.1 (left). Polthier's discretization (see [17]) is similar to the one given by Desbrun et al (see [8]). He takes $w_{ij} = \frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij})$, without imposing the normalization condition $\sum w_{ij} = 1$.

It is easy to see that the discretization (2.7) could not be an approximation of $\Delta_{\mathcal{M}}$, since $\Delta_M p_i \to 0$ as the size of the surface mesh goes to zero.

2. Mayer's Discretization (see [15], 2001).
Discretizing (3.4) at p_i over the triangular surface mesh M, Mayer got the following approximation.

$$\Delta_M^{(2)} f(p_i) = \frac{1}{\mathcal{A}(p_i)} \sum_{j \in N_1(i)} \frac{\|p_{j_-} - p_j\| + \|p_{j_+} - p_j\|}{2} \frac{f(p_j) - f(p_i)}{\|p_j - p_i\|}, \tag{2.8}$$

where $\mathcal{A}(p_i)$ is the sum of areas of triangles around p_i.

3. Desbrun et al's discretization (see [8], 1999, [9], 2000).
From a differential geometry definition of mean curvature normal, one has

$$\lim_{diam(\mathcal{A})\to 0} \frac{3\nabla\mathcal{A}}{2\mathcal{A}} = -H(p), \tag{2.9}$$

where $\mathcal{A}$ is the area of a small region around the point p where the curvature is needed, and ∇ is the gradient with respect to the (x, y, z) coordinates of p. From (2.9), Desbrun et al get the following discretization

$$\Delta_M^{(3)} f(p_i) = \frac{3}{\mathcal{A}(p_i)} \sum_{j\in N_1(i)} \frac{\cot\alpha_{ij} + \cot\beta_{ij}}{2} [f(p_j) - f(p_i)], \tag{2.10}$$

where α_{ij} and β_{ij} are defined as before. (2.10) could be easily derived from (2.9) by writing $\mathcal{A}(p_i)$ in the following form

$$\mathcal{A}(p_i) = \sum_{j\in N_1(i)} \frac{1}{2}\sqrt{\|p_j - p_i\|^2 \|p_{j_+} - p_i\|^2 - (p_j - p_i, p_{j_+} - p_i)^2},$$

and then taking partials of $\mathcal{A}(p_i)$ with respect to the coordinates of p_i.

4. Meyer et al's discretization (see [16],2002).

$$\Delta_M^{(4)} f(p_i) = \frac{1}{\mathcal{A}_M(p_i)} \sum_{j\in N_1(i)} \frac{\cot\alpha_{ij} + \cot\beta_{ij}}{2} [f(p_j) - f(p_i)],$$

where $\mathcal{A}_M(p_i)$ is an area for vertex p_i as shown in Fig 2.1 (right), where q_j is the circumcenter point for the triangle $[p_{j\_} p_j p_i]$ if the triangle is non-obtuse. If the triangle is obtuse, q_j is chosen to be the midpoint of the edge opposite to the obtuse angle.

The discretizations $\Delta_M^{(k)} f$, $k = 1, \cdots, 4$ have been reviewed in [29]. It has been shown that all of these discretizations are not convergent in the general cases. Only two of them, which are proposed by Desbrun et al and Meyer et al, converge for some special cases. Now we repeat the convergent results as follows:

Theorem 2.1 *Let M be a triangulation of surface $\mathcal{M}$. Let p_i be a vertex of M with valence six, and let p_j be its neighbor vertices for $j \in N_1(i)$. Suppose p_i and p_j are on a sufficiently smooth parametric surface $G(\xi_1, \xi_2) \in I\!R^3$, and there exist q_i, q_j such that*

$$p_i = G(q_i) \;\; p_j = G(q_j) \;\; \text{and} \;\; q_j = q_{j_-} + q_{j_+} - q_i, \quad j \in N_1(i).$$

Let f be a sufficiently smooth function over surface $\mathcal{M}$. Then we have

$$\lim_{h\to 0} \frac{3}{\mathcal{A}(p_i, h)} \sum_{j\in N_1(i)} \frac{\cot\alpha_{ij}(h) + \cot\beta_{ij}(h)}{2} [f(p_j(h)) - f(p)] = \Delta_{\mathcal{M}} f(p),$$
$$\lim_{h\to 0} \frac{1}{\mathcal{A}_M(p_i, h)} \sum_{j\in N_1(i)} \frac{\cot\alpha_{ij}(h) + \cot\beta_{ij}(h)}{2} [f(p_j(h)) - f(p)] = \Delta_{\mathcal{M}} f(p),$$

where $p_j(h) = G(q_j(h))$, $q_j(h) = q + h(q_j - q)$, $j \in N_1(i)$, and $\mathcal{A}(p_i, h)$, $\mathcal{A}_M(p_i, h)$, $\alpha_{ij}(h)$ and $\beta_{ij}(h)$ are defined as before from vertices $p_j(h)$.

Note that if the domain of the surface $G(\xi_1, \xi_2)$ is triangulated by the three directional partition (see Fig. 5.1(a)), then the condition of the theorem is satisfied.

3 Our Discretization of LBOs

In this section we propose several other alternatives of the discretization of LBO. Let $\mathcal{M}$ be a Riemannian manifold. Let $\partial\mathcal{M}$ be the the C^∞ boundary of $\mathcal{M}$. Let n be the unit outward normal vector field to the boundary, and X be a C^1 vector field on $\mathcal{M}$ with compact support. Then (see [14], page 330)

$$\int_{\mathcal{M}} (\mathrm{div}_{\mathcal{M}} X) dv_{\mathcal{M}} = \int_{\partial\mathcal{M}} \langle X, n \rangle dv_{\partial\mathcal{M}}, \tag{3.1}$$

where $dv_{\mathcal{M}}$ and $dv_{\partial\mathcal{M}}$ denote the canonical metric on $\mathcal{M}$ and $\partial\mathcal{M}$, respectively. Let f be a C^2 smooth function on $\mathcal{M}$, then $\nabla_{\mathcal{M}} f$ is a C^1 vector field on $\mathcal{M}$. Let $X = \nabla_{\mathcal{M}} f$ in (3.1). Then, since $\Delta_{\mathcal{M}} f = \mathrm{div}_{\mathcal{M}} \nabla_{\mathcal{M}} f$, we have

$$\int_{\mathcal{M}} \Delta_{\mathcal{M}} f dv_{\mathcal{M}} = \int_{\partial\mathcal{M}} \langle \nabla_{\mathcal{M}} f, n \rangle dv_{\partial\mathcal{M}}. \tag{3.2}$$

3.1 Indirect discretization

Suppose M be a triangular discretization of $\mathcal{M}$. Let p_i be the i-th vertex of M. Then (3.2) could be approximately discretized as

$$\Delta_M f(p_i) = \frac{1}{2\mathcal{A}(p_i)} \sum_{j \in N_1(i)} n_j^T \left[\nabla_{\mathcal{M}} f(p_j) + \nabla_{\mathcal{M}} f(p_{j_+}) \right] \|p_j - p_{j_+}\|, \tag{3.3}$$

where $\mathcal{A}_i$ is the sum of the areas of the triangles surrounding to p_i and n_j is the unit outward normal of the edge $[p_j p_{j_+}]$. Let $\bar{n}_j := n_j \|p_j - p_{j_+}\|$. Then it is easy to verify that

$$\bar{n}_j = \frac{\tilde{n}_j}{2A_j} \quad \text{with} \quad \tilde{n}_j = -[(p_i - p_j, p_j - p_{j_+})(p_{j_+} - p_i) + (p_i - p_{j_+}, p_{j_+} - p_j)(p_j - p_i)],$$

where A_j is the area of the triangle $[p_i p_j p_{j_+}]$. In discretization (3.3), gradient vectors are used. These gradients need to be discretized further (see Section 4). We therefore call (3.3) as indirect discretization. For this discretization, we have the following convergent result.

Theorem 3.1 *Let p_i be a vertex of M with valence n. Suppose p_i and p_j are on a sufficiently smooth parametric surface $G(\xi_1, \xi_2) \in I\!R^3$, for all $j \in N_1(i)$, and there exist q_i, $q_j \in I\!R^2$ such that*

$$det[q_{j_-} - q_i, q_j - q_i] * det[q_{j_+} - q_i, q_j - q_i] < 0, \quad j \in N_1(i),$$

$$p_i = G(q_i), \quad p_j = G(q_j), \; j \in N_1(i).$$

Let f be a smooth function on surface G. Then

$$\lim_{h\to 0} \frac{1}{2\mathcal{A}(p_i, h)} \sum_{j \in N_1(i)} n_j(h)^T \left[\nabla_{\mathcal{M}} f(p_j(h)) + \nabla_{\mathcal{M}} f(p_{j_+}(h)) \right] \|p_j(h) - p_{j_+}(h)\| = \Delta_{\mathcal{M}} f(p_i),$$

where $p_j(h) = G(q_j(h))$, $q_j(h) = q_i + h(q_j - q_i)$ for $j \in N_1(i)$.

The proof of the theorem is meticulous, we put it into the Appendix. Let $\nabla_M f$ be a discretization of $\nabla_{\mathcal{M}} f$. Then it is easy from the proof of Theorem 3.1 to obtain the following conclusion:

Corollary 3.1 *Under the condition of Theorem 3.1, if*

$$\nabla_M f(p_j(h)) = \nabla_{\mathcal{M}} f(p_j(h)) + O(h^2), \qquad j \in N_1(i),$$

then

$$\lim_{h\to 0} \frac{1}{2\mathcal{A}(p_i, h)} \sum_{j \in N_1(i)} n_j(h)^T \left[\nabla_M f(p_j(h)) + \nabla_M f(p_{j_+}(h)) \right] \|p_j(h) - p_{j_+}(h)\| = \Delta_{\mathcal{M}} f(p_i).$$

3.2 Direct Discretization

A. Discretization via Gauss Formula

Since (3.2) could be written as

$$\int_{\mathcal{M}} \Delta_{\mathcal{M}} f dv_{\mathcal{M}} = \int_{\partial\mathcal{M}} \partial_n f dv_{\partial\mathcal{M}}, \tag{3.4}$$

we can derive the following discretization

$$\Delta_M^{(D)} f(p_i) = \frac{1}{\mathcal{A}(p_i)} \sum_{j\in N_1(i)} \frac{\bar{f}_j - \bar{f}'_j}{\|\bar{n}_j - \bar{n}'_j\|} \|p_j - p_{j_+}\|, \tag{3.5}$$

where

$$\bar{n}_j = -\frac{(p_i - p_j, p_j - p_{j_+})(p_{j_+} - p_i) + (p_i - p_{j_+}, p_{j_+} - p_j)(p_j - p_i)}{2A_j}, \tag{3.6}$$

$$\bar{n}'_j = -\frac{(p'_j - p_j, p_j - p_{j_+})(p_{j_+} - p'_j) + (p'_j - p_{j_+}, p_{j_+} - p_j)(p_j - p'_j)}{2A'_j}, \tag{3.7}$$

$$\bar{f}_j = -\frac{(p_i - p_j, p_j - p_{j_+})(f_{j_+} - f_i) + (p_i - p_{j_+}, p_{j_+} - p_j)(f_j - f_i)}{2A_j}, \tag{3.8}$$

$$\bar{f}'_j = -\frac{(p'_j - p_j, p_j - p_{j_+})(f_{j_+} - f'_j) + (p'_j - p_{j_+}, p_{j_+} - p_j)(f_j - f'_j)}{2A'_j}, \tag{3.9}$$

A_j and A'_j are the areas of the triangles $[p_ip_jp_{j_+}]$ and $[p'_jp_jp_{j_+}]$, respectively, $f_j = f(p_j)$, p'_j is the opposite vertex of p_i to the edge $[p_jp_{j_+}]$, and $f'_j = f(p'_j)$. Note that n_j and n'_j are vectors perpendicular to the edge $[p_jp_{j_+}]$ with length $\|p_j - p_{j_+}\|$, and in the triangles $[p_ip_jp_{j_+}]$ and $[p'_jp_jp_{j_+}]$, respectively. Hence, $\frac{\bar{f}_j - \bar{f}'_j}{\|\bar{n}_j - \bar{n}'_j\|}$ is an approximation of $\partial_n f$ on the edge $[p_jp_{j_+}]$.

Theorem 3.2 *Let p_i be a vertex of M with valence n. Suppose p_i, p_j and p'_j are on a sufficiently smooth parametric surface $G(\xi_1, \xi_2) \in I\!R^3$, for all $j \in N_1(i)$, and there exist q_i, q_j and q'_j in $I\!R^2$ such that*

$$q_j + q_{j_+} = q_i + q'_j \quad j \in N_1(i),$$

$$p_i = G(q_i), \quad p_j = G(q_j), \quad p'_j = G(q'_j) \quad j \in N_1(i).$$

Let f be a smooth function on surface G. Then

$$\lim_{h\to 0} \frac{1}{\mathcal{A}(p_i, h)} \sum_{j\in N_1(i)} \frac{\bar{f}_j(h) - \bar{f}'_j(h)}{\|\bar{n}_j(h) - \bar{n}'_j(h)\|} \|p_j(h) - p_{j_+}(h)\| = \Delta_{\mathcal{M}} f(p_i), \tag{3.10}$$

where $\bar{n}_j(h)$, $\bar{n}'_j(h)$, $\bar{f}_j(h)$ and $\bar{f}'_j(h)$ are defined as (3.6)–(3.9) using $p_j(h) = G(q_j(h)), p'_j(h) = G(q'_j(h))$ with $q_j(h) = q_i + h(q_j - q_i)$, $q'_j(h) = q_i + h(q'_j - q_i)$ for $j \in N_1(i)$.

Proof. We can derive that under the condition of the theorem

$$\frac{\bar{f}_j(h) - \bar{f}'_j(h)}{\|\bar{n}_j(h) - \bar{n}'_j(h)\|} \|p_j(h) - p_{j_+}(h)\| = \bar{n}_j(h)^T \frac{\nabla_{\mathcal{M}} f(p_j(h)) + \nabla_{\mathcal{M}} f(p_{j_+}(h))}{2} + O(h^3).$$

Then the convergence result (3.10) follows from the proof of Theorem 3.1.

B. Discretization via Quadratic Fitting

Now we use a biquadratic fit of the surface data and function data to calculate the approximate LBO. Let p_i be a vertex of M with valence n, p_j be its neighbor vertices for $j \in N_1(i)$, and assume that $[p_ip_jp_{j_+}]$ are the neighbor triangles of p_i. Then the biquadratic fit is computed as follows:

1. Compute angles

$$\alpha_k = cos^{-1}[(p_{i_k} - p_i, p_{i_{k+1}} - p_i)/\|p_{i_k} - p_i\|\|p_{i_{k+1}} - p_i\|], \quad k = 1, \cdots, n,$$

and then compute the angles $\beta_k = 2\pi\alpha_k / \sum_{j=1}^n \alpha_j$

2. Set $q_0 = (0, 0)$ and

$$q_k = \|p_{i_k} - p_i\|(cos\theta_k, sin\theta_k), \quad \theta_k = \beta_1 + \cdots + \beta_{k-1}, \quad k = 1, \cdots, n \ (\theta_1 = 0).$$

3. Take the basis functions $\{B_l(\xi_1, \xi_2)\}_{l=0}^5 = \{1, \xi_1, \xi_2, \frac{1}{2}\xi_1^2, \xi_1\xi_2, \frac{1}{2}\xi_2^2\}$, and determine the coefficient $c_l \in I\!R^3$ so that

$$\sum_{l=0}^5 c_l B_l(q_k) = p_{i_k}, \quad k = 0, \cdots, n \ (\text{assume} \ \ i_0 = i)$$

in the lease square sense. This system is solved by solving the normal equation. Let $A = (B_l(q_k))_{k=0,l=0}^{n,5} \in I\!R^{(n+1)\times 6}$, and let $C = (A^T A)^{-1} A^T \in I\!R^{6\times(n+1)}$, then

$$[c_0, \cdots, c_5]^T = C[p_{i_0}, \cdots, p_{i_n}]^T.$$

4. Let

$$[d_0, \cdots, d_5]^T = C[f(p_{i_0}), \cdots, f(p_{i_n})]^T.$$

Then compute LBO of $\tilde{f} = \sum_{l=0}^5 d_l B_k(\xi_1, \xi_2)$ over the surface $\tilde{G} = \sum_{l=0}^5 c_l B_l(\xi_1, \xi_2)$ at $(0, 0)$, using the formula (2.4). We denote this approximate LBO as $\Delta_M^{(F)} f(p_i)$, where the superscript "F" stands for "fitting". It is easy to see that

$$t_1 = c_1, \quad t_2 = c_1, \quad t_{11} = c_3, \quad t_{12} = c_4, \quad t_{22} = c_5.$$

Denote the second, third, fourth, fifth and sixth rows of C as C_1, C_2, C_{20}, C_{11} and C_{02}, respectively, then we can see that

$$\begin{aligned} \frac{\partial f}{\partial \xi_j} &= C_j[f(p_{i_0}), \cdots, f(p_{i_n})]^T, \quad j = 1, 2, \\ \frac{\partial^2 f}{\partial \xi_j \xi_k} &= C_{jk}[f(p_{i_0}), \cdots, f(p_{i_n})]^T, \quad j + k = 2. \end{aligned}$$

Substituting these quantities into (2.4), we will get an approximation of LBO as

$$\Delta_M^{(F)} f(p_i) = \sum_{k=0}^n w_k f(p_{i_k}). \tag{3.11}$$

Note that the coefficients w_k depend only on the geometric data of the mesh M.

The construction algorithm above may fail in the following two cases. **a**. The system is under-determined in the case $n = 3$ or $n = 4$. **b**. The coefficient matrix of the normal equation is singular or nearly singular. For case **a**, we will replace the basis functions by $\{B_l(\xi_1, \xi_2)\}_{l=1}^5 = \{1, \xi_1, \xi_2, \frac{1}{2}(\xi_1^2 + \xi_2^2)\}$, and solve the fitting problem in a lower dimensional space. For case **b**, we look for a least square solution with minimal normal. Let $A^T A x = b$ be the linear system in the matrix form. We find a least square solution such that $\|x\|_2 = min$. Such a solution could be computed by the SVD decomposition of $A^T A$ (see [11], Chapter 5).

4 Discretization of Gradient

The discretization (3.3) of the LBO in the last section requires the gradient vector of f at each vertex. Hence we need to discretize the gradient further. In this section, we propose two simple approaches for discretizing the gradient.

4.1 Discretization via Linear Approximation

Let $T_j = [p_i p_j p_{j+}]$ be a triangle adjacent to vertex p_i. Then by a linear interpolation of the surface and function on the surface, we can derive that the gradient can be approximated on the triangle by

$$\begin{aligned}\nabla_{T_j} f = & \frac{1}{4A_j^2} \{ f_i[(p_i - p_j, p_j - p_{j_+})(p_{j_+} - p_i) + (p_i - p_{j_+}, p_{j_+} - p_j)(p_j - p_i)] \\ & + f_j[(p_j - p_i, p_i - p_{j_+})(p_{j_+} - p_j) + (p_j - p_{j_+}, p_{j_+} - p_i)(p_i - p_j)] \\ & + f_{j_+}[(p_{j_+} - p_j, p_j - p_i)(p_i - p_{j_+}) + (p_{j_+} - p_i, p_i - p_j)(p_j - p_{j_+})] \} ,\end{aligned} \tag{4.1}$$

where A_j denotes the area of T_j. Having approximate gradients on triangles, the gradient at a vertex p_i can be approximated by a weighted average of the gradients on the surrounding triangles of p_i:

$$\nabla_M^{(A)} f(p_i) = \frac{1}{\mathcal{A}(p_i)} \sum_{j \in N_1(i)} A_j \nabla_{T_j} f, \tag{4.2}$$

where $\mathcal{A}(p_i) = \sum_{j \in N_1(i)} A_j$. The superscript "A" of $\nabla_M^{(A)}$ stands for "averaging".

Theorem 4.1 *Under the conditions of Theorem 3.1, we have*

$$\nabla_M^{(A)} f(p_i) = \nabla_{\mathcal{M}} f(p_i) + O(h). \tag{4.3}$$

Furthermore, if

$$n = 2m, \quad q_{i_{k+m}}(h) = q_i - h(q_{i_k} - q_i) \quad \text{for} \quad k = 1, 2, \cdots, m, \tag{4.4}$$

then

$$\nabla_M^{(A)} f(p_i) = \nabla_{\mathcal{M}} f(p_i) + O(h^2). \tag{4.5}$$

See Appendix for the proof of the theorem.

4.2 Discretization via Loop's Subdivision

It follows from (2.2) that, the computation of gradient involves the computation of the surface tangents t_1, t_2 and partials $\frac{\partial f}{\partial \xi_1}$, $\frac{\partial f}{\partial \xi_2}$ under a local parameterization of the surface. Now we compute these quantities from the limit surface $\tilde{G}$ and the limit function $\tilde{f}$ of the Loop's subdivision for the triangular surface mesh M and the function f on the surface. We denote these tangents and partials by $\tilde{t}_1, \tilde{t}_2$, $\frac{\partial \tilde{f}}{\partial \xi_1}$, $\frac{\partial \tilde{f}}{\partial \xi_2}$. At a vertex p_i with surrounding vertices p_{i_j}, $i_j \in N_1(i)$, the tangent directions corresponding to the edge $[p_i p_{i_j}]$ is given by (see [3])

$$\begin{aligned}\tilde{t}_k &= cos\frac{2\pi(k-1)}{n} a_1^0 + sin\frac{2\pi(k-1)}{n} a_{n-1}^0 \\ &= \frac{2}{n} \sum_{j=1}^{n} cos\frac{2\pi(j-k)}{n} p_{i_j}, \quad k = 1, 2, \cdots, n,\end{aligned}$$

where

$$a_1^0 = \frac{2}{n} \sum_{j=1}^{n} cos\frac{2\pi(j-1)}{n} p_{i_j}, \ a_{n-1}^0 = \frac{2}{n} \sum_{j=1}^{n} sin\frac{2\pi(j-1)}{n} p_{i_j}.$$

Similarly, partials of $\tilde{f}$ corresponding to the edge $[p_i p_{i_j}]$ is given by

$$\frac{\partial f}{\partial \xi_k} = \frac{2}{n} \sum_{j=1}^{n} cos\frac{2\pi(j-k)}{n} f(p_{i_j}), \quad k = 1, 2, \cdots, n.$$

Therefore, we get an approximation of $\nabla_{\mathcal{M}}$ as follows

$$\nabla_M^{(L)} f(p_i) = [p_{i_1}, \cdots, p_{i_n}][V_1,\ V_2]G_M^{-1}[V_1,\ V_2]^T[f(p_{i_1}), \cdots, f(p_{i_n})]^T,$$

where

$$V_k = \frac{2}{n}\left[cos\frac{2\pi(1-k)}{n}, cos\frac{2\pi(2-k)}{n}, \cdots, cos\frac{2\pi(n-k)}{n}\right]^T \in I\!R^n, \quad k=1,2,$$

and $G_M = [\tilde{t}_1,\ \tilde{t}_2]^T[\tilde{t}_1,\ \tilde{t}_2]$. Note that

$$\sum_{j=1}^{n} cos\frac{2\pi(j-k)}{n} = 0, \quad k = 1, \cdots, n, \tag{4.6}$$

and $\nabla_M^{(L)} f(p_i)$ does not depend on vertex p_i and function value $f(p_i)$.

Theorem 4.2 *Under the conditions of Theorem 3.1, we have*

$$\nabla_M^{(L)} f(p_i) = \nabla_{\mathcal{M}} f(p_i) + O(h). \tag{4.7}$$

Furthermore, if $n = 2m$, and the condition (4.4) holds, then

$$\nabla_M^{(L)} f(p_i) = \nabla_{\mathcal{M}} f(p_i) + O(h^2) \tag{4.8}$$

Again, we put the proof of this theorem into the Appendix.

4.3 Remarks on the Discretizations of gradient and LBO

We have proposed two simple schemes for computing the approximate gradient. Both the average gradient $\nabla_M^{(A)} f(p_i)$ from linear interpolation and the gradient $\nabla_M^{(L)} f(p_i)$ from Loop's subdivision have close forms, therefore they are easy to use and easy to compute.

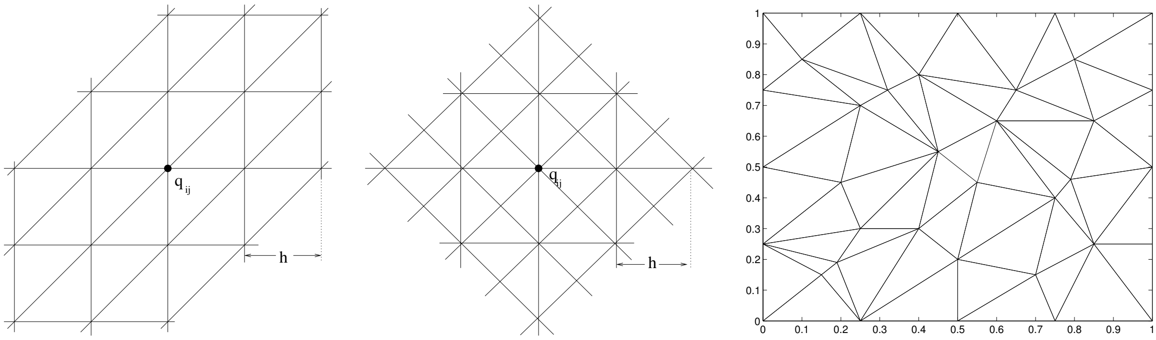
Both the approximate gradients have linear convergent rate. For a special case, where the valence of a vertex p_i is an even number and the domain triangulation has certain symmetric property (see (4.4)), $\nabla_M^{(A)} f(p_i)$ and $\nabla_M^{(L)} f(p_i)$ have quadratic convergent rate, and therefore *the resulting discrete LBOs, denoted as $\Delta_M^{(A)} f(p_i)$ and $\Delta_M^{(L)} f(p_i)$, are convergent.* In many applications, the condition (4.4) could be satisfied. For instance, suppose we have a sequence of hierarchical triangular surface mesh generated by Loop's subdivision, conditions for quadratic convergence are satisfied at each regular vertex.

The direct discretization $\Delta_M^{(D)} f(p_i)$ is also simple and has a close form, and it converges under another condition. This discretization as well as $\nabla_M^{(A)} f(p_i)$ and $\nabla_M^{(L)} f(p_i)$ involve two-ring neighbor vertices of p_i. We call the collection of the involved vertices of a discrete LBO as its *support*. Hence, $\nabla_M^{(A)} f(p_i)$, $\nabla_M^{(L)} f(p_i)$ and $\Delta_M^{(D)} f(p_i)$ have larger supports.

The discrete LBO $\Delta_M^{(F)} f(p_i)$ obtained from the biquadratic fitting requires to solve a 6×6 linear system in the least square sense at each vertex. Hence it is not as simple as the other three. However, $\Delta_M^{(F)} f(p_i)$ converges in general, except for the vertices with valence $n = 3, 4$. Furthermore, this discretization involves one-ring neighbor vertices rather than two-ring. If two-ring vertices are used, the convergence is guaranteed even for $n = 3, 4$.

Comparing these discrete LBOs, we can see that from approximation power point of view, $\Delta_M^{(F)}$ is the best, $\Delta_M^{(2)}$ is the worst. From simplicity point of view $\Delta_M^{(3)}$ and $\Delta_M^{(4)}$ are the best, while $\Delta_M^{(F)}$ is the worst. Others are good under some special conditions. Depending on the natural of the application problem to be solved, one may choose a proper convergent discrete LBO to achieve one's goal. If none of the $\Delta_M^{(3)}$, $\Delta_M^{(4)}$, $\Delta_M^{(A)}$, $\Delta_M^{(L)}$ and $\Delta_M^{(D)}$ satisfy the required convergent condition, we at least have $\Delta_M^{(F)}$ in hand.

The convergence results are established under various special conditions. However, these special cases are very useful and important, because many numerical simulations of geometric partial differential



(a) (b) (c)

Fig 5.1: The triangulation of the domain. (a) Three directional triangular partition. (b). Four directional triangular partition. (c) Unstructured triangular partition.

equations are conducted over a triangulated domain formed by a uniform three-directional partition or four-directional partition (see Fig. 5.1) (a) and (b)). The three-directional partition satisfies the conditions of all the convergence theorems. The four-directional partition satisfies the conditions of Theorem 4.1 and 4.2.

Finally, we point out that the given conditions in each of the convergence theorems are sufficient only. This means that there may be other conditions under which the discrete LBOs converge. The problem searching for necessary and sufficient conditions for the convergence of these discrete LBOs is left open.

5 Numerical Experiments

The aim of this section is to exhibit the numerical behaviors of the discrete LBOs defined in Section 2 and 3. To show the numerical convergence of the discrete LBOs, we take several two variable functions:

$$\begin{aligned} F_1(x,y) &= \sqrt{4-(x-0.5)^2-(y-0.5)^2}, \\ F_2(x,y) &= \tanh(9y-9x), \\ F_3(x,y) &= \frac{1.25+\cos(5.4y)}{6+6(3x-1)^2}, \\ F_4(x,y) &= \exp\left(-\frac{81}{16}\left[(x-0.5)^2+(y-0.5)^2\right]\right). \end{aligned}$$

over xy-plane as three dimensional surfaces so that the exact mean curvatures can be easily computed. Both the exact and approximated mean curvatures are computed at some selected domain points $q_{ij} = (x_i, y_j)$. As a first case, these points are chosen as $(x_i, y_j) = (\frac{i}{20}, \frac{j}{20})$ for $i = 1, \cdots, 19,\ j = 1, \cdots, 19$. The surfaces are triangulated around q_{ij} by triangulating the domain first, and then mapping the planner triangulation onto the surfaces by the selected bivariate functions. The domain around q_{ij} is triangulated locally in two different ways as shown in Fig. 5.1(a), 5.1(b), to illustrate how the domain triangulation affect the convergence. The second case we consider is that we choose an unstructured domain triangulation as shown in Fig. 5.1(c). For observing the convergence/non-convergence property, finer and finer domain triangulations are generated. For case (a) and (b) in Fig. 5.1, h are taken to be $2^{-3}, 2^{-5}, 2^{-7}, \cdots$. For case (c), the domain is recursively subdivided by the bisection linear subdivision. Hence, $h = h_0, h_0/2, h_0/4, \cdots$, where $h_0 = 0.3354$ is the maximal value of the edge lengths of the triangulation as shown in Fig. 5.1(c).

The experiments show that as $h \to 0$, the maximal error of the approximated mean curvature approaches to Ch^k for a constant C and a certain k. For example, for the domain triangulation as shown in Fig 5.1(a) and function F_1, the maximal error of the approximated mean curvature computed by (2.8) and the exact mean curvature computed from the continuous surfaces is as follows: 0.36362, 0.36356, 0.36356, 0.36355, 0.36355, 0.36355, 0.36355, 0.36355, 0.36355, 0.36355 for $h = 2^{-3}, 2^{-5}, 2^{-7}, \cdots, 2^{-21}$. Table

5.1 shows the asymptotic values of the maximal error of the approximated mean curvature computed by discrete LBOs and the exact mean curvature computed from the continuous surfaces for the domain triangulation as shown in Fig. 5.1(a). This domain triangulation satisfies the conditions of Theorem 2.1, 3.2, 4.1 and 4.2. Hence convergence property is observed for $\Delta^{(3)}$, $\Delta^{(4)}$, $\Delta^{(A)}$, $\Delta^{(L)}$, $\Delta^{(D)}$ and, of course, $\Delta^{(F)}$. Furthermore, the convergence rates are quadratic.

Table 5.1: The maximal Errors for domain (a)

F_i	$\Delta^{(2)}$	$\Delta^{(3)}$	$\Delta^{(4)}$	$\Delta^{(A)}$	$\Delta^{(L)}$	$\Delta^{(F)}$	$\Delta^{(D)}$
F_1	3.64e–1	1.23e–1*h^2	1.19e–1*h^2	4.35e–2*h^2	1.63e–1*h^2	9.30e–2*h^2	8.99e–2*h^2
F_2	4.66e+0	4.34e+2*h^2	4.34e+2*h^2	1.16e+3*h^2	9.59e+2*h^2	2.85e+2*h^2	8.92e+2*h^2
F_3	3.68e+0	2.03e+2*h^2	1.38e+2*h^2	7.12e+2*h^2	5.77e+2*h^2	4.90e+1*h^2	5.07e+2*h^2
F_4	6.13e+0	7.19e+2*h^2	6.79e+2*h^2	2.14e+3*h^2	1.51e+3*h^2	1.37e+2*h^2	1.42e+3*h^2

Table 5.2 shows the asymptotic value of the maximal error for the domain triangulation as shown in Fig. 5.1(b). This domain triangulation satisfies the conditions of Theorem 4.1 and 4.2. Hence convergence property is observed for $\Delta^{(A)}$, $\Delta^{(L)}$ and $\Delta^{(F)}$. An exceptional case is that $\Delta^{(4)}$ converges for the surface defined by F_2. Though the conditions of Theorem 3.2 is not satisfied directly, $\Delta^{(D)}$ converges, the reason is that if we merge the two triangles near q_{ij} in each quadrant into one, then we can see that the condition of Theorem 3.2 is really satisfied.

Table 5.2: The maximal Errors for domain (b)

F_i	$\Delta^{(2)}$	$\Delta^{(3)}$	$\Delta^{(4)}$	$\Delta^{(A)}$	$\Delta^{(L)}$	$\Delta^{(F)}$	$\Delta^{(D)}$
F_1	3.96e–1	2.51e–1	1.64e–2	3.12e–2*h^2	5.55e–2*h^2	5.57e–2*h^2	1.56e–2*h^2
F_2	4.23e+0	2.11e+0	4.34e+2*h^2	1.18e+3*h^2	9.60e+2*h^2	2.85e+2*h^2	1.12e+3*h^2
F_3	4.07e+0	2.58e+0	5.49e–1	3.90e+2*h^2	3.16e+2*h^2	9.86e+1*h^2	2.42e+2*h^2
F_4	6.68e+0	4.39e+0	1.06e+0	1.19e+3*h^2	8.26e+2*h^2	1.59e+2*h^2	7.75e+2*h^2

Table 5.3 shows the asymptotic value of the maximal error for the domain triangulation as shown in Fig. 5.1(c). This domain triangulation does not satisfy the conditions of Theorem 2.1, 4.1 and 4.2. Hence no convergence property is observed for $\Delta^{(3)}$, $\Delta^{(4)}$, $\Delta^{(A)}$ and $\Delta^{(L)}$. But approximation property is observed for these discrete operators. However, $\Delta^{(2)}$ has no approximation property (error increase in the rate $O(h^{-1})$). The linear subdivision of the domain makes the conditions of Theorem 3.2 be satisfied, hence convergent property is observed for $\Delta^{(D)}$. This is an interesting case, because for Loop's surface subdivision scheme, the domain of the Loop's surface is undergoing linear subdivision. Therefore, $\Delta^{(D)}$ over the Loop's subdivision surface mesh will converge at the each ordinary vertex as the subdivision process is repeated. In computing the maximal error of $\Delta^{(F)}$ we have excluded a vertex which is near the origin, because this vertex has valence 4. $\Delta^{(F)}$ will not converge at this point.

Table 5.3: The maximal Errors for domain (c)

F_i	$\Delta^{(2)}$	$\Delta^{(3)}$	$\Delta^{(4)}$	$\Delta^{(A)}$	$\Delta^{(L)}$	$\Delta^{(F)}$	$\Delta^{(D)}$
F_1	1.01e+0*h^{-1}	4.23e–1	9.81e–2	1.78e–1	1.91e–1	2.26e–2*h^2	9.11e–3*h
F_2	1.01e+0*h^{-1}	5.07e+0	4.80e+0	7.82e–1	1.21e+0	8.99e+0*h	1.46e+1*h
F_3	1.02e+0*h^{-1}	9.60e–1	3.72e–1	4.13e–1	4.55e–1	3.18e–1*h	8.15e+0*h
F_4	1.27e+0*h^{-1}	3.52e+0	1.30e+0	8.98e–1	8.51e–1	8.32e–1*h	1.52e+1*h

Finally, we illustrate how the supports of the discrete LBOs affect approximation errors. Table 5.4 shows the maximal error for the domain triangulation as shown in Fig. 5.1(a) with a fixed triangulation ($h = 2^{-5}$). Exact mean curvature at each point is computed from the given function. The approximated mean curvature is computed from discrete data. But we perturb randomly the discrete function value with 0.1%, to show that the discrete LBOs with larger supports are insensitive to the higher frequency errors. The results in the table show that the discrete LBOs with larger supports usually give better results. Time costs (in second) for computing the data in Table 5.4 are summarized in Table 5.5. The computation is conducted on a Dell PC equipped with an Intel(R) CPU (1.90GHz).

Table 5.4: The maximal Errors for domain (a) with $h = 2^{-6}$

F_i	$\Delta^{(2)}$	$\Delta^{(3)}$	$\Delta^{(4)}$	$\Delta^{(A)}$	$\Delta^{(L)}$	$\Delta^{(F)}$	$\Delta^{(D)}$
F_1	3.697	7.158	7.489	0.912	1.058	7.081	2.850
F_2	5.719	3.010	3.263	1.049	0.945	2.603	1.083
F_3	3.522	0.635	0.505	0.643	0.531	0.591	0.625
F_4	7.395	2.539	2.353	1.827	1.484	1.887	1.872

Table 5.5: Time costs for the computation of Table 5.4

F_i	$\Delta^{(2)}$	$\Delta^{(3)}$	$\Delta^{(4)}$	$\Delta^{(A)}$	$\Delta^{(L)}$	$\Delta^{(F)}$	$\Delta^{(D)}$
F_1	0.025	0.026	0.041	0.068	0.052	0.032	0.028
F_2	0.026	0.029	0.042	0.074	0.054	0.036	0.032
F_3	0.031	0.032	0.047	0.082	0.059	0.038	0.034
F_4	0.025	0.031	0.042	0.072	0.053	0.035	0.029

6 Applications of Discrete LBOs

An obvious application of the discrete LBOs is use them to compute approximated mean curvatures from a triangulated surface as we did in the last section. We have illustrate that the discrete LBOs with larger supports usually works better for noisy data. One of our main purposes for proposing these discrete LBOs is for solving geometric partial differential equations, such as numerical simulation of various geometric flows (mean curvature flow, surface diffusion flow, Willmore flow etc.), surface smoothing, surface construction and surface image processing. In the following, we give a few examples that show the applications in these problems. We refer the interested readers to [30] for detail descriptions of various geometric PDEs and how these PDEs are solved with given boundary conditions.

Simulation of Geometric Flows. The aim of the simulation of the geometric flow is to see how the surface evolves under the flow. Fig. 6.1 show some simple examples of the simulation of the mean curvature flow, the averaged mean curvature flow, surface diffusion flow and Willmore flow with the input four pipes serving as boundary constraints (Fig. 6.1(a)). We use the solutions of these geometric flows to blend the input four pipes. $\Delta_{\mathcal{M}}$ in these flows is approximated by $\Delta_M^{(F)}$. Fig. 6.1(b) shows an initial blending mesh construction of the pipes which defines the topology of evolved surface and serves as an initial condition. (c), (d), (e) and (f) are numerical solutions of the mean curvature flow, averaged mean curvature flow, the surface diffusion flow and the Willmore flow, respectively. All these solutions are obtained after 100 iterations with time step length 0.001. The solution of the mean curvature flow at this stage is still undergoing rapid change, further evolution will lead to a pinch-off of the surface. The solutions of the other three flows are almost stable at this moment.

Surface Hole Filling. Given a surface mesh with a hole, we construct a fair surface to fill the hole with specified geometric continuity on the boundary. Fig 6.1(g)–(i) show such an example, where a head mesh with a hole at the nose is given (figure (g)) with G^1 continuity requirement. An initial G^0 construction of the nose is shown in (h) using the method in [1] with some noise added. The fair filling surface (figures (i)) are generated using the surface diffusion flow. $\Delta_{\mathcal{M}}$ in the flow is discretized as $\Delta_M^{(F)}$.

Surface Smoothing. Given a surface mesh with noise, now we use the following mean curvature flow to smooth the surface:

$$\frac{\partial x}{\partial t} = -a(x)\Delta_{\mathcal{M}}(x),$$

where $a(x) > 0$ is a function which is adaptive to the mesh density. We choose it as A_i/A at vertex p_i, where A is the average of all A_i. Fig. 6.1(j) shows an input noisy mesh. (k) is the smoothing result after 6 iterations using discrete LBO $\Delta_M^{(A)}$ with time step-length 0.001. (l) is the smooth result of (k)

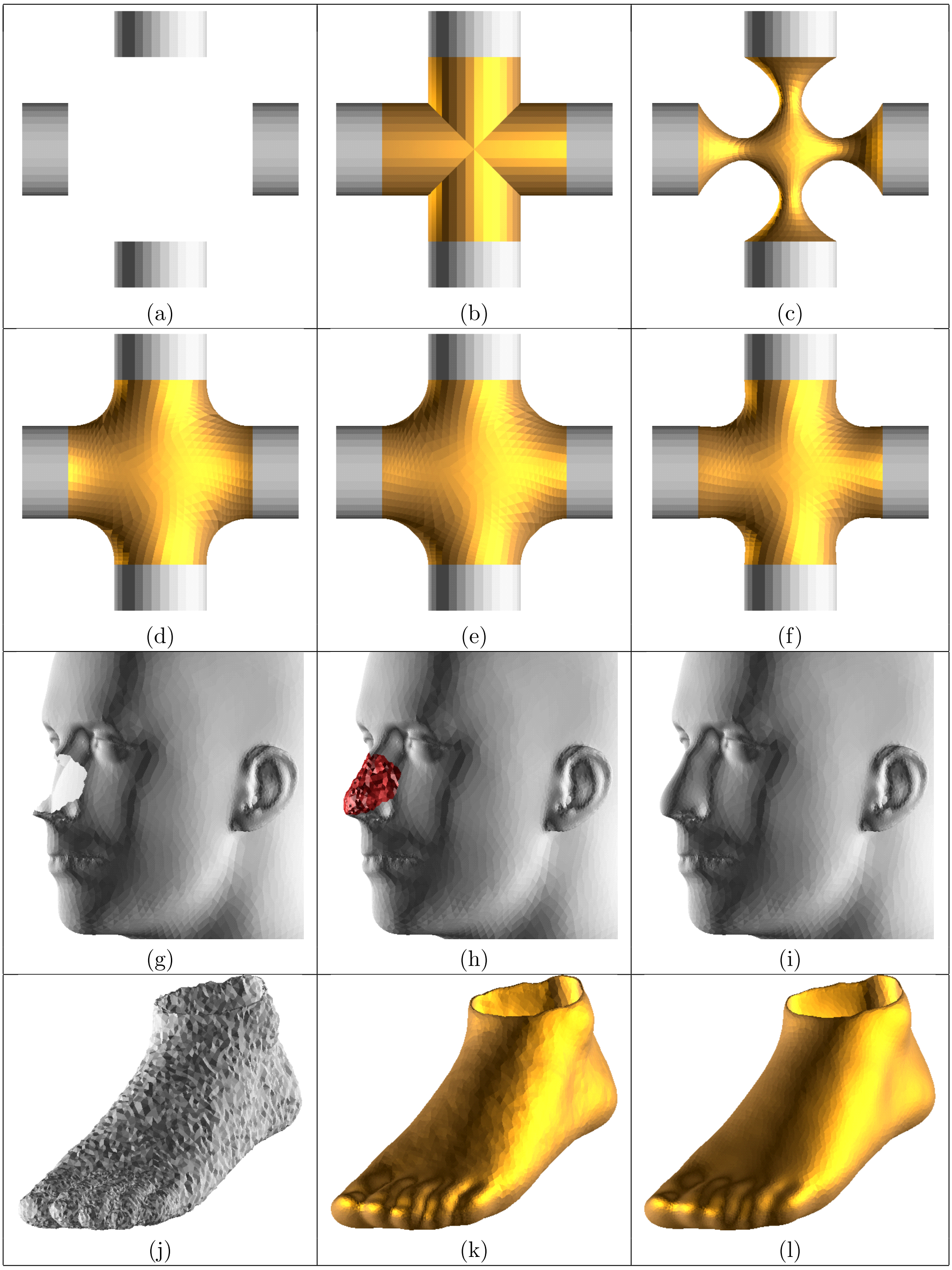


Fig 6.1: (a) shows four input pipes which serve as boundary constraints of the evolving surface. (b) shows an initial blending mesh construction of the pipes. (c), (d), (e) and (f) show numerical solutions of the mean curvature flow, the averaged mean curvature flow, the surface diffusion flow and the Willmore flow, respectively. All these solutions are obtained after 100 iterations with time step length 0.001. (g) shows a head mesh with a hole around the nose. (h) shows an initial filler construction of the brandy nose. (i) is the faired filler surface, after 1 iteration, generated using surface diffusion flow. The time step length is chosen to be 0.0001. Fig. (j), (k) and (l) show the denoising effect of discrete LBOs, where (j) shows the input. (k) and (l) are the smoothing results after 6 and 12 iterations with time step-length 0.001.

after another 6 iterations using discrete LBO $\Delta_M^{(F)}$ with the same time step-length. Since $\Delta_M^{(A)}$ has larger support, it will affect low frequency noise and insensitive to the higher frequency error. Hence the combination use of the discrete LBOs with different sizes of support can yield more desirable results. The deliberated use of these discrete LBOs on denoising is beyond the scope of this paper. We shall report our research results on this aspect elsewhere.

7 Conclusions

We have proposed several discretization schemes for LBO on the triangular surfaces. The presented numerical and application examples show that these discrete LBOs can be applied in solving geometric PDEs or surface processing, or to compute the approximate values of LBO acting on discrete function on surface. Convergence results under some specified conditions are established and these theoretical results are verified by numerical examples. We also show that the discrete LBOs with larger supports are insensitive to higher frequency errors. Hence they have antinoise property when applying them to noise data.

8 Appendix

The proof Theorem 3.1. For simplifying the notation in this proof, we assume $i = 0$ and $N_1(i) = \{1, 2, \cdots, n\}$. Let

$$d_j = (c_j, s_j) := (cos\theta_j, sin\theta_j) := (q_j - q_0)/\|q_j - q_0\|.$$

and assume that $\theta_1 > \theta_2 > \cdots > \theta_n$. Then

$$q_j(h) = q_0 + hw_j d_j$$

with $w_j = \|q_j - q_0\|$. Now we compute $A_j(h)$ and $\bar{n}_j(h)$. Since

$$4A_j(h)^2 = \|p_j(h) - p_0\|^2\|p_{j+1}(h) - p_0\|^2 - (p_j(h) - p_0, p_{j+1}(h) - p_0)^2,$$

$$\tilde{n}_j(h) = -[(p_0 - p_j(h), p_j(h) - p_{j+1}(h))(p_{j+1}(h) - p_0) + (p_0 - p_{j+1}(h), p_{j+1}(h) - p_j(h))(p_j(h) - p_0)],$$

we can derive that

$$4A_j(h)^2 = w_j^2 w_{j+1}^2 (c_{j+1}s_j - c_j s_{j+1})^2 (g_{11}g_{22} - g_{12}^2)h^4 + O(h^5).$$

Since

$$c_{j+1}s_j - c_j s_{j+1} = sin(\theta_j - \theta_{j+1}) > 0,$$

we have

$$2A_j(h) = w_j w_{j+1}(c_{j+1}s_j - c_j s_{j+1})\sqrt{g}h^2 + O(h^3)$$

and

$$\tilde{n}_j(h) = w_j w_{j+1}(c_{j+1}s_j - c_j s_{j+1})h^3 g[t_1, t_2]G^{-1}S_j + O(h^4)$$

with $S_j = \begin{bmatrix} s_j w_j - s_{j+1}w_{j+1} \\ c_{j+1}w_{j+1} - c_j w_j \end{bmatrix}$. Hence

$$\bar{n}_j(h) = \tilde{n}_j(h)/(2A_j(h)) = h\sqrt{g}[t_1, t_2]G^{-1}S_j + O(h^2).$$

Now we compute $\nabla_{\mathcal{M}} f(p_j(h))$ by Taylor expansion. It follows from (2.2) that

$$\begin{aligned}
\nabla_{\mathcal{M}} f(p_j(h)) &= \nabla_{\mathcal{M}} f(p_0) + hw_j D_{d_j}([t_1,t_2]G^{-1}\nabla_{\mathcal{M}} f(p))\big|_{p=p_0} + O(h^2) \\
&= \nabla_{\mathcal{M}} f(p_0) + hw_j D_{d_j}([t_1,t_2])G^{-1}\nabla f(p_0)) \\
&+ \ hw_j[t_1,t_2]D_{d_j}[G^{-1}\nabla f(p)]\big|_{p=p_0} + O(h^2) \\
&= \nabla_{\mathcal{M}} f(p_0) + hw_j[c_j t_{11} + s_j t_{12}, c_j t_{12} + s_j t_{22}]G^{-1}\nabla f(p_0)) \\
&+ \ hw_j[t_1,t_2]D_{d_j}[G^{-1}\nabla f(p)]\big|_{p=p_0} + O(h^2).
\end{aligned}$$

Hence

$$\begin{aligned}
&\frac{\nabla_{\mathcal{M}} f(p_j(h)) + \nabla_{\mathcal{M}} f(p_{j+1}(h))}{2} = \nabla_{\mathcal{M}} f(p_0) \\
&\quad + \frac{h}{2}\left\{(w_j c_j + w_{j+1}c_{j+1})[t_{11},t_{12}] + (w_j s_j + w_{j+1}s_{j+1})[t_{12},t_{22}]\right\} G^{-1}\nabla f(p_0) \\
&\quad + \frac{h}{2}[t_1,t_2](w_j D_{d_j} + w_{j+1}D_{d_{j+1}})[G^{-1}\nabla f(p)]\Big|_{p=p_0} + O(h^2),
\end{aligned}$$

and therefore

$$\bar{n}_j(h)^T \frac{\nabla_{\mathcal{M}} f(p_j(h)) + \nabla_{\mathcal{M}} f(p_{j+1}(h))}{2} = h\sqrt{g}S_j^T G^{-1}[t_1,t_2]^T\nabla f(p_0) \tag{8.1}$$

$$+ \frac{h^2}{2}\sqrt{g}S_j^T G^{-1}[(w_j c_j + w_{j+1}c_{j+1})G_1 + (w_j s_j + w_{j+1}s_{j+1})G_2]G^{-1}\nabla f(p_0) \tag{8.2}$$

$$+ \frac{h^2}{2}\sqrt{g}(w_j D_{d_j} + w_{j+1}D_{d_{j+1}})S_j^T[G^{-1}\nabla f(p)]\Big|_{p=p_0} + O(h^3), \tag{8.3}$$

where

$$G_1 = \begin{bmatrix} g_{111} & g_{112} \\ g_{211} & g_{212} \end{bmatrix}, \quad G_2 = \begin{bmatrix} g_{112} & g_{122} \\ g_{212} & g_{222} \end{bmatrix}.$$

Now we consider the sum of $\bar{n}_j(h)^T \frac{\nabla_{\mathcal{M}} f(p_j(h)) + \nabla_{\mathcal{M}} f(p_{j+1}(h))}{2}$. Since $\sum_{j=1}^n S_j^T = 0$, the sum of right-handed side of (8.1) is zero. The sum of the term (8.2) is

$$\frac{h^2\sqrt{g}}{2}\left[\sum_{j=1}^n S_j^T(w_j c_j + w_{j+1}c_{j+1})G^{-1}G_1 + \sum_{j=1}^n S_j^T(w_j s_j + w_{j+1}s_{j+1})G^{-1}G_2\right] G^{-1}\nabla f(p_0).$$

Since

$$\begin{aligned}
\sum_{j=1}^n S_j^T(w_j c_j + w_{j+1}c_{j+1}) &= \sum_{j=1}^n \begin{bmatrix} w_j^2 c_j s_j - w_{j+1}^2 c_{j+1}s_{j+1} + w_j w_{j+1}(c_{j+1}s_j - c_j s_{j+1}) \\ (c_{j+1}w_{j+1})^2 - (c_j w_j)^2 \end{bmatrix}^T \\
&= [\alpha, 0],
\end{aligned}$$

and similarly

$$\sum_{j=1}^n S_j^T(w_j s_j + w_{j+1}s_{j+1}) = [0,\alpha]$$

with $\alpha = \sum_{j=1}^n w_j w_{j+1}(c_{j+1}s_j - c_j s_{j+1})$, the sum of (8.2) is

$$\frac{h^2\alpha}{2\sqrt{g}}\begin{bmatrix} g_{11}g_{212} + g_{22}g_{111} - g_{12}(g_{211} + g_{112}) \\ g_{11}g_{222} + g_{22}g_{112} - g_{12}(g_{212} + g_{122}) \end{bmatrix}^T G^{-1}\nabla f(p_0). \tag{8.4}$$

Since $\mathcal{A}(p_i,h) = \sum A_j(h) = \frac{h^2\alpha\sqrt{g}}{2}$, dividing (8.4) by $\mathcal{A}(p_i,h)$, we obtain the first term of the right-handed side of (2.5). Now we compute the sum of the first term of (8.3). Since

$$w_j D_{d_j} + w_{j+1} D_{d_{j+1}} = \left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2}\right] \tilde{S}_j,$$

where $\tilde{S}_j = [w_j c_j + w_{j+1} c_{j+1}, w_j s_j + w_{j+1} s_{j+1}]^T$. The sum of the first term of (8.3) is

$$\begin{aligned}
&\frac{h^2\sqrt{g}}{2}\left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2}\right] \sum_{j=1}^{n} \tilde{S}_j S_j^T [G^{-1}\nabla_{\mathcal{M}} f(p)]\Bigg|_{p=p_0} \\
&\quad = \frac{h^2\sqrt{g}}{2}\left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2}\right] \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix} [G^{-1}\nabla_{\mathcal{M}} f(p)]\Bigg|_{p=p_0} \\
&\quad = \frac{h^2\alpha\sqrt{g}}{2}\left[\frac{\partial}{\partial \xi_1}, \frac{\partial}{\partial \xi_2}\right] [G^{-1}\nabla_{\mathcal{M}} f(p)]\Bigg|_{p=p_0}
\end{aligned}$$

Dividing this term by $\mathcal{A}(p_i,h)$, we obtain the last term of (2.5). Therefore, the theorem is proved. ◇

The Proof of Theorem 4.1. Consider $\nabla_{T_j} f$. Let

$$q_j(h) = q_i + w_j h (c_j, s_j)^T, \quad j \in N_1(i).$$

Then we can derive that

$$4A_j(h)^2 = w_j^2 w_{j_+}^2 (c_j s_{j_+} - c_{j_+} s_j)^2 \det(G)\ h^4 + a_5^{(j)} h^5 + O(h^6),$$

where $a_5^{(j)}$ is a constant. The second factor of the right-handed side of (4.1) is

$$\begin{aligned}
& w_j^2 w_{j_+}^2 (c_j s_{j_+} - c_{j_+} s_j)^2 h^4 \left[(g_{22}t_1 - g_{12}t_2)\frac{\partial f}{\partial \xi_1} + (g_{11}t_2 - g_{12}t_1)\frac{\partial f}{\partial \xi_2}\right] + b_5^{(j)} h^5 + O(h^6) \\
= \quad & w_j^2 w_{j_+}^2 (c_j s_{j_+} - c_{j_+} s_j)^2 h^4 [t_1, t_2] \begin{bmatrix} g_{22} & -g_{12} \\ -g_{12} & g_{11} \end{bmatrix} \nabla f + b_5^{(j)} h^5 + O(h^6).
\end{aligned}$$

Hence,

$$\nabla_{T_j} f = \nabla_{\mathcal{M}} f(p_i) + C_1^{(j)} h + O(h^2), \tag{8.5}$$

where $C_1^{(j)}$ is a constant. Taking a weighted average of $\nabla_{T_j} f$ with weight $\frac{A_j(h)}{\sum_{j\in N_1(i)} A_j(h)}$, we obtain (4.3). Now we prove (4.5). Under the condition (4.4), it is easy to see that

$$a_5^{(i_{k+m})} = -a_5^{(i_k)}, b_5^{(i_{k+m})} = -b_5^{(i_k)}, C_1^{(i_{k+m})} = -C_1^{(i_k)}, \quad k = 1, \cdots, m.$$

The coefficient of h in (4.3) is therefore cancelled. Hence (4.5) is derived. ◇

The Proof of Theorem 4.2. Let $q_j(h) = q_i + w_j h d_j$ with $d_j = (c_j, s_j)^T$, $j \in N_1(i)$. Then we can derive that

$$\begin{aligned}
\tilde{t}_l &= \frac{2}{n}\sum_{k=1}^{n} cos\frac{2\pi(k-l)}{n}\left[p_i + w_{i_k} h D_{d_{i_k}} G + \frac{1}{2} w_{i_k}^2 h^2 D_{d_{i_k}}^2 G + O(h^3)\right] \\
&= \frac{2}{n}\sum_{k=1}^{n} cos\frac{2\pi(k-l)}{n}\left[w_{i_k} h D_{d_{i_k}} G + \frac{1}{2} w_{i_k}^2 h^2 D_{d_{i_k}}^2 G + O(h^3)\right] \\
&= [t_1, t_2][A_{1l}, A_{2l}]^T h + D_{2l} h^2 + O(h^3),
\end{aligned} \tag{8.6}$$

where A_{1l} and A_{2l} are constants, $D_{2l} \in I\!R^3$ is a constant vector. Similarly

$$\frac{\partial \tilde{f}}{\partial \xi_l} = \left[\frac{\partial f}{\partial \xi_1}, \frac{\partial f}{\partial \xi_2}\right] [A_{1l}, A_{2l}]^T h + E_{2l} h^2 + O(h^3), \tag{8.7}$$

where E_{2l} is constant. From Lemma 2.1, we obtain (4.7). Now we assume $n = 2m$ and (4.4) holds. Since

$$cos\frac{2\pi(k-l)}{n} = -cos\frac{2\pi(k-l+m)}{n}, \quad k = 1, \cdots, m,$$

the coefficient D_{2l} and E_{2l} in (8.6) and (8.7) are zero. Therefore, (4.8) holds. $\diamond$

9 Acknowledgment

I would like to thank C. L. Bajaj for many useful discussions concerning the importance and utility of the discrete Laplace-Beltrami operators.

References

[1] C. Bajaj and I. Ihm. Algebraic surface design with hermite interpolation. *ACM Transactions on Graphics*, 19(1):61–91, 1992.

[2] C. Bajaj and G. Xu. Anisotropic Diffusion of Surface and Functions on Surfaces. *ACM Transaction on Graphics*, 22(1):4–32, 2003.

[3] C. Bajaj, G. Xu, and J. Warren. Acoustics Scattering on Arbitrary Manifold Surfaces. In *Proceedings of Geometric Modeling and Processing, Theory and Application, Japan*, pages 73–82, 2002.

[4] M. Bertalmio, G. Sapiro, L. T. Cheng, and S. Osher. A framework for solving surface partial differential equations for computer graphics applications. CAM Report 00-43, UCLA, Mathematics Department, 2000.

[5] M. P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, New Jersey, 1976.

[6] X. Chen and F. Schmitt. Intrinsic surface properties from surface triangulation. In Proceedings, European Conference on Computer Vision, pages 739–743, 1992.

[7] U. Clarenz, U. Diewald, and M. Rumpf. Anisotropic Geometric Diffusion in Surface Processing. In *Proceedings of Viz2000, IEEE Visualization*, pages 397–505, Salt Lake City, Utah, 2000.

[8] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow. *SIGGRAPH99*, pages 317–324, 1999.

[9] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Discrete Differential-Geometry Operators in nD. In *Proc. VisMath'02*, Berlin, Germany, 2002.

[10] K. Fujiwara. Eigenvalues of laplacians on a closed riemannian manifold and its nets. In Proceedings of the AMS, pages 2585–2594, 1995.

[11] G. Golub and C. Van Loan. Matrix Computations. The Johns Hopkins University Press, 1996.

[12] B. Hamann. Curvature approximation for triangulated surfaces. In G. Farin et al, editor, *Geometric Modelling*, pages 139–153. Springer-Verlag, 1993.

[13] R. Kimmel, R. Malladi, and N. Sochen. Image Processing via the Beltrami Operator. In Proc. of 3-rd Asian Conf. on Computer Vision, Hong Kong, January 8-11, 1998.

[14] S. Lang. Differential and Riemannian Manifolds. Springer-Verlag, 1995.

[15] U. F. Mayer. Numerical Solutions for the Surface Diffusion Flow in Three Space Dimensions. *Computational and Applied Mathematics (to appear)*, 2001.

[16] M. Meyer, M. Desbrun, P. Schröder, and A. Barr. Discrete Differential- Geometry Operator for Triangulated 2-manifolds. In *Proc. VisMath'02*, Berlin, Germany, 2002.

[17] K. Polthier. Computational Aspects of Discrete Minimal Surfaces. In *Proc. of the Clay Summer School on Global Theory of Minimal Surfaces, J. Hass, D. Hoffman, A. Jaffe, H. Rosenberg, R. Schoen, M. Wolf (Eds.), to appear*, 2002.

[18] B. Haar Romeny. *Geometry Driven Diffusion in Computer Vision*. Boston, MA: Kluwer, 1994.

[19] S. Rosenberg. *The Laplacian on a Riemannian Manifold*. Cambridge, Uviversity Press, 1997.

[20] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge, University Press, 2001.

[21] R. Schneider and L. Kobbelt. Generating Fair Meshes with G^1 Boundary conditions. In *Geometric Modeling and Processing*, pages 251–261. 2000.

[22] R. Schneider and L. Kobbelt. Geometric Fairing of Triangular Meshes for Free-form Surface Design, 2001.

[23] G. Simonett. The Willmore Flow for Near Spheres. *Differential and Integral Equations*, 14(8):1005–1014, 2001.

[24] G. Taubin. Estimating the tensor of curvatures of a surface from a polyhedral approximation. In Proceedings 5th Intl. Conf. on Computer Vision (ICCV'95), pages 902–907, 1995.

[25] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH '95 Proceedings*, pages 351–358, 1995.

[26] G. Taubin. Signal processing on polygonal meshes. In *EUROGRAPHICS*. 2000.

[27] J. Weickert. *Anisotropic Diffusion in Image Processing.* B. G. Teubner Stuttgart, 1998.

[28] T. J. Willmore. *Riemannian Geometry.* Clarendon Press, 1993.

[29] G. Xu. Convergence of Discrete Laplace-Beltrami Operators over Surfaces. Research Report No. ICM-03-014, Institute of Computational Mathematics, Chinese Academy of Sciences, 2003.

[30] G. Xu, Q. Pan, and C. Bajaj. *Discrete Surface Modeling using Geometric Flows.* TICAM Report 03-38, August 2003, Institute for Computational Engineering and Sciences, The University of Texas at Austin, 2003.