

G^1 Spline Surface Construction By Geometric Partial Differential Equations Using Mixed Finite Element Methods *

Guoliang Xu ¹⁾ Chandrajit Bajaj ²⁾

¹⁾ State Key Laboratory of Scientific and Engineering Computing
Institute of Computational Mathematics, Academy of Mathematics and
System Sciences, Chinese Academy of Sciences, Beijing 100190, China

²⁾ Department of Computer Sciences and Institute of Computational
Engineering & Sciences, University of Texas at Austin, Austin TX 78712

Abstract

Variational formulations of three fourth order geometric partial differential equations are derived, and based on which mixed finite element methods are presented for constructing G^1 smooth B-spline surfaces. Solutions to several surface modeling problems, including surface fairing, free-form surface design, surface blending and hole filling with G^1 continuity, are handled by this approach. Experimental results show that our method is efficient and gives the desired results. Convergence properties of the proposed method are additionally investigated, which justify that the method is both effective and mathematically correct.

Key words: Spline surface, Discretizations, GPDE, Convergence.

MR (2000) Classification: 65D17

1 Introduction

A surface satisfying a geometric partial differential equation (GPDE) is referred to as a GPDE surface (see [27]). GPDE surfaces are favorable in the areas of computer graphics and computer aided geometric design since they often possess certain optimal properties. For instance, the GPDE surface which is a steady state solution of mean curvature flow has minimal surface area (see [17]), and the Willmore surface which is a steady solution of Willmore flow has minimal total squared mean curvature (see [11, 12]). However, the construction of GPDE surfaces is not a trivial task, because GPDEs are highly nonlinear in general. In recent years, the constructions of discrete GPDE surfaces (typically using triangular meshes) have been studied intensively using divided difference-like discretization techniques (see [22, 24, 26, 27]) or even finite element methods (see [21, 23]). However, the construction techniques for continuous GPDE surfaces,

*Project support in part by NSFC grant 60773165 and National Key Basic Research Project of China (2004CB318000).

using Bernstein-Bézier surface, B-spline surfaces and NURBS surfaces (see [14]), are few and far between???

As a subset of NURBS surfaces and an extension of Bernstein-Bézier surfaces, the class of B-spline surfaces has become the most popular for shape design and geometric modeling. B-spline surfaces are widely used in the fields of computer graphics, animation and computer added design (CAD) and computer added manufacturing (CAM). The aim of this paper is to establish an efficient, reliable and mathematically correct method for constructing GPDE B-spline surfaces with specified G^1 boundary conditions, and using mixed finite element methods.

Divided difference-like techniques are generally more efficient and easier to implement. However, they lack of??? convergence analysis. In contrast, the finite element methods, while computationally more intensive, have sound and well established mathematical theory (see [20]). More importantly, finite element method can handle G^1 boundary condition much more naturally than the divided difference-like method. In divided difference-like method, the length of the tangent vectors on the surface boundaries have to be taken into account, while in the finite element method presented in this paper, the lengths of the tangent vectors have no effect on the constructed surfaces, and hence is our solution method of choice.

The rest of the paper is organized as follows: Section 2 introduces necessary background material from differential geometry and our earlier publications (for completeness). Section 3 describes the used nonlinear GPDEs, their variational formulations and the details of the discretizations of the variational forms in both the spatial and the temporal directions. Details of the implementation and examples to illustrate the different effects achievable from the solutions of these GPDEs are given in section 4.

2 Preliminaries and Notations

This section introduce the necessary background material and notations, including definitions and relations amongst a few geometric differential operators, curvatures and spline functions.

2.1 Differential Geometry of Parametric Surface

Let $\mathcal{S} := \{\mathbf{x}(u^1, u^2) \in \mathbb{R}^3 : (u^1, u^2) \in \mathcal{D} \subset \mathbb{R}^2\}$ be a parametric surface. For simplicity, we assume it is sufficiently smooth and orientable. Let $g_{\alpha\beta} = \langle \mathbf{x}_{u^\alpha}, \mathbf{x}_{u^\beta} \rangle$ and $b_{\alpha\beta} = \langle \mathbf{n}, \mathbf{x}_{u^\alpha u^\beta} \rangle$ be the coefficients of the first and the second fundamental forms of \mathcal{S} with

$$\begin{aligned} \mathbf{x}_{u^\alpha} &= \frac{\partial \mathbf{x}}{\partial u^\alpha}, \quad \mathbf{x}_{u^\alpha u^\beta} = \frac{\partial^2 \mathbf{x}}{\partial u^\alpha \partial u^\beta}, \quad \alpha, \beta = 1, 2, \\ \mathbf{n} &= (\mathbf{x}_u \times \mathbf{x}_v) / \|\mathbf{x}_u \times \mathbf{x}_v\|, \quad (u, v) := (u^1, u^2), \end{aligned}$$

where $\langle \cdot, \cdot \rangle$, $\|\cdot\|$, $\cdot \times \cdot$ stand for inner product, Euclidean norm and cross product in \mathbb{R}^3 respectively. Let

$$[g^{\alpha\beta}] = [g_{\alpha\beta}]^{-1}, \quad [b^{\alpha\beta}] = [b_{\alpha\beta}]^{-1}, \quad g = \det[g_{\alpha\beta}], \quad b = \det[b_{\alpha\beta}].$$

Curvatures. To introduce the notions of mean curvature and Gaussian curvature, we use the concept of *Weingarten map* or *shape operator* (see [9]). The shape operator of surface \mathcal{S} is a self-adjoint linear map on the tangent space $T_{\mathbf{x}}\mathcal{S} := \text{span}\{\mathbf{x}_u, \mathbf{x}_v\}$ defined by

$$\mathcal{W} : T_{\mathbf{x}}\mathcal{S} \rightarrow T_{\mathbf{x}}\mathcal{S},$$

such that

$$\mathcal{W}(\mathbf{x}_u) = -\mathbf{n}_u, \quad \mathcal{W}(\mathbf{x}_v) = -\mathbf{n}_v. \quad (2.1)$$

We can easily represent this linear map by a 2×2 matrix $S = [b_{\alpha\beta}][g^{\alpha\beta}]$. In particular,

$$[\mathbf{n}_u, \mathbf{n}_v] = -[\mathbf{x}_u, \mathbf{x}_v]S^T \quad (2.2)$$

is valid. The eigenvalues k_1 and k_2 of S are *principal curvatures* of \mathcal{S} and their arithmetic average and product are the *mean curvature* H and the *Gaussian curvature* K , respectively. That is

$$H = \frac{k_1 + k_2}{2} = \frac{\text{tr}(S)}{2}, \quad K = k_1 k_2 = \det(S).$$

Let $\mathbf{H} = H\mathbf{n}$ and $\mathbf{K} = K\mathbf{n}$, they are referred to as mean curvature vector and Gaussian curvature vector, respectively. Now we introduce a few geometric differential operators.

Tangential gradient operator. Suppose $f \in C^1(\mathcal{S})$ then the tangential gradient operator ∇ acting on f is defined as

$$\nabla f = [\mathbf{x}_u, \mathbf{x}_v][g^{\alpha\beta}][f_u, f_v]^T \in \mathbb{R}^3. \quad (2.3)$$

For a vector-valued function $\mathbf{f} = [f_1, \dots, f_k]^T \in C^1(\mathcal{S})^k$, we define its gradient by

$$\nabla \mathbf{f} = [\nabla f_1, \dots, \nabla f_k] \in \mathbb{R}^{3 \times k}.$$

It is easy to see that

$$\nabla \mathbf{x} = [\mathbf{x}_u, \mathbf{x}_v][g^{\alpha\beta}][\mathbf{x}_u, \mathbf{x}_v]^T, \quad (2.4)$$

$$\nabla \mathbf{n} = -[\mathbf{x}_u, \mathbf{x}_v][g^{\alpha\beta}]S[\mathbf{x}_u, \mathbf{x}_v]^T, \quad (2.5)$$

and both $\nabla \mathbf{x}$ and $\nabla \mathbf{n}$ are symmetric 3×3 matrices.

The second tangential operator. Let $f \in C^1(\mathcal{S})$. Then the second tangential operator \diamond acting on f is defined as

$$\diamond f = [\mathbf{x}_u, \mathbf{x}_v][h_{\alpha\beta}][f_u, f_v]^T \in \mathbb{R}^3. \quad (2.6)$$

where

$$[h_{\alpha\beta}] = \frac{1}{g} \begin{bmatrix} b_{22} & -b_{12} \\ -b_{12} & b_{11} \end{bmatrix}. \quad (2.7)$$

The third tangential operator. Let $f \in C^1(\mathcal{S})$. Then the third tangential operator \circlearrowleft acting on f is defined as

$$\circlearrowleft f = [\mathbf{x}_u, \mathbf{x}_v][g^{\alpha\beta}]S[f_u, f_v]^T \in \mathbb{R}^3.$$

The three tangential operators introduced above are in the tangent space $T_{\mathbf{x}}\mathcal{S}$, they are linearly dependent. In fact, we have the following Lemma.

Lemma 2.1 *For any function $f \in C^1(\mathcal{S})$, we have*

$$2H\nabla f = \circlearrowleft f + \diamond f. \quad (2.8)$$

Proof. From the definitions of the three tangential operators we know that if the equality

$$2H[g^{\alpha\beta}] = [g^{\alpha\beta}]S + [h_{\alpha\beta}]. \quad (2.9)$$

holds, then (2.8) is obviously valid. From a direct calculation, (2.9) could be easily derived.

Divergence operator. Suppose \mathbf{v} is a smooth vector field on surface \mathcal{S} , then the divergence operator div_s acting on \mathbf{v} is defined as

$$\text{div}_s(\mathbf{v}) = \frac{1}{\sqrt{g}} \left[\frac{\partial}{\partial u}, \frac{\partial}{\partial v} \right] \left[\sqrt{g} [g^{\alpha\beta}] [\mathbf{x}_u, \mathbf{x}_v]^T \mathbf{v} \right]. \quad (2.10)$$

Laplace-Beltrami operator. Let $f \in C^2(\mathcal{S})$. Then the Laplace-Beltrami operator (LBO) Δ acting on f is defined as (see [9], p. 83)

$$\Delta f = \text{div}_s(\nabla f).$$

Obviously, Δ is a second order differential operator. It is well known that LBO relates to the mean curvature vector via the equation: $\Delta \mathbf{x} = 2\mathbf{H}$.

Let us introduce a divergence theorem (see [3], p. 142).

Lemma 2.2 (Riemannian Divergence Theorem) *Let \mathcal{S} be an orientable surface, Ω a subregion of \mathcal{S} with a piecewise smooth boundary $\partial\Omega$. Let $\mathbf{n}_c \in T_{\mathbf{x}}\mathcal{S}$ ($\mathbf{x} \in \partial\Omega$) be the outward unit normal along the boundary $\partial\Omega$. Then for any given C^1 vector field \mathbf{v} on \mathcal{S} , we have*

$$\int_{\Omega} \text{div}_s(\mathbf{v})dA = \int_{\partial\Omega} \langle \mathbf{v}, \mathbf{n}_c \rangle ds.$$

Theorem 2.1 (Green's formula for LBO) *Let \mathcal{S} be an orientable surface, Ω a subregion of \mathcal{S} with a piecewise smooth boundary $\partial\Omega$. Let $\mathbf{n}_c \in T_{\mathbf{x}}\mathcal{S}$ ($\mathbf{x} \in \partial\Omega$) be the outward unit normal along the boundary $\partial\Omega$. Then for a given C^1 smooth vector field \mathbf{v} on \mathcal{S} , we have*

$$\int_{\Omega} [\langle \mathbf{v}, \nabla f \rangle + f \text{div}(\mathbf{v})]dA = \int_{\partial\Omega} f \langle \mathbf{v}, \mathbf{n}_c \rangle ds. \quad (2.11)$$

Proof. Taking \mathbf{v} as $f\mathbf{v}$ in the Riemannian divergence theorem, we immediately obtain (2.11).

2.2 Spline surface

There are several equivalent approaches to define spline functions, including divided difference of truncated power function (see [7, 18]), the blossoming method (see [15]) and Cox-Door's recursive formulas (see [6, 8]). We adopt the approach of recursive formulas as these are the easiest to program.

Definition. Given a positive integer m , nonnegative integer k and a knot sequence

$$u_0 \leq \cdots \leq u_i \leq u_{i+1} \leq u_{i+2} \leq \cdots \leq u_{m+2k}.$$

$U = \{u_0, \dots, u_{m+2k}\}$ is referred to as knot vector. Then B-splines basis functions are defined as follows

$$\left\{ \begin{array}{l} N_{i,0}(u) = \begin{cases} 1, & \text{for } u \in [u_i, u_{i+1}), \\ 0, & \text{otherwise,} \end{cases} \quad i = 0, 1, \dots, m + 2k - 1, \\ N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} N_{i,k-1}(u) + \frac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} N_{i+1,k-1}(u), \quad i = 0, 1, \dots, m + k - 1, \\ \text{Assume } \frac{0}{0} = 0 \end{array} \right. \quad (2.12)$$

where i is the index of $N_{i,k}(u)$, k is the degree.

Spline surface. For given positive integers m, n and a nonnegative integer k , and knot vectors

$$U = \{u_0, \dots, u_{m+2k}\}, \quad V = \{v_0, \dots, v_{n+2k}\},$$

the degree k four-sided spline surface is defined as

$$\mathbf{x}(u, v) = \sum_{i=0}^{m+k-1} \sum_{j=0}^{n+k-1} \mathbf{p}_{ij} N_{i,k}(u) N_{j,k}(v), \quad (u, v) \in \Omega := [0, 1]^2,$$

where $\mathbf{p}_{ij} \in \mathbb{R}^3$ are called control points of surface $\mathbf{x}(u, v)$. If $i = 0$ or $m, j = 0$ or n , \mathbf{p}_{ij} is called boundary control points. Other control points are called inner control points. In order to have the spline surface be at least C^2 smooth, we take $k \geq 3$.

3 Construction GPDE Spline Surfaces

This section is devoted to details of the construction of B-spline GPDE surfaces, including the formulation of GPDEs, and their variational forms.

3.1 GPDEs and their Variational Forms

To construct G^1 smooth GPDE B-spline surface patch, we use three fourth order equations, namely surface diffusion flow (SDF), Willmore flow (WF) and quasi-surface flow (QSDF).

Surface Diffusion Flow

$$\frac{\partial \mathbf{x}}{\partial t} = -2\Delta H \mathbf{n}. \quad (3.1)$$

This flow is introduced by Mullins in 1957 (see [13]), to describe the interface motion law of growing crystal.

Willmore flow

$$\frac{\partial \mathbf{x}}{\partial t} = - [\Delta H + 2H(H^2 - K)] \mathbf{n}. \quad (3.2)$$

There are now sound published research papers that use this flow (see [2, 11, 12, 19]). If the initial surface is a sphere, Willmore flow keeps the spherical shape unchanged.

Quasi-surface diffusion flow

$$\frac{\partial \mathbf{x}}{\partial t} = -\Delta^2 \mathbf{x}. \quad (3.3)$$

This flow was introduced in [24], and is used in surface design. It is known that tangent flow on a surface does not alter the surface shape (see [10]). Hence if we remove the tangential flow portion of (3.3), we obtain a geometric flow

$$\frac{\partial \mathbf{x}}{\partial t} = -2(\Delta H - 4H^3 + 2HK) \mathbf{n}.$$

If \mathcal{S} is a closed surface and A stands for its area, then by Green's formula we obtain (see [4, 17] for the rate of change rates of the surface area and the enclosed volume of the evolved surface)

$$\frac{d}{dt} A(t) = -2 \int_{\mathcal{S}(t)} [\|\nabla H\|^2 + 2H^2(2H^2 - K)] dA \leq 0.$$

Hence, Quasi-surface diffusion flow is area diminishing. The shrinkage stops when $H \equiv 0$.

Next we present variational form formulations for the GPDE equations (3.1)–(3.3). The detailed derivations are given in the appendix.

The mixed variational form of (3.1) is: Find $(\mathbf{x}, \mathbf{y}) \in H^2(\mathcal{S})^3 \times H^1(\mathcal{S})^3$ such that

$$\begin{cases} \int_{\mathcal{S}} \frac{\partial \mathbf{x}}{\partial t} \phi \, dA + 2 \int_{\mathcal{S}} [\phi \otimes \mathbf{y} - \mathbf{n}(\nabla \phi)^T \nabla \mathbf{y}] \mathbf{n} \, dA = \mathbf{0}, & \forall \phi \in H_0^1(\mathcal{S}), \\ \int_{\mathcal{S}} \mathbf{y} \psi \, dA + \frac{1}{2} \int_{\mathcal{S}} (\nabla \mathbf{x})^T \nabla \psi \, dA - \frac{1}{2} \int_{\partial \mathcal{S}} \mathbf{n}_c \psi \, ds = \mathbf{0}, & \forall \psi \in H^1(\mathcal{S}), \\ \mathcal{S}(0) = \mathcal{S}_0, \quad \partial \mathcal{S}(t) = \Gamma, \quad \mathbf{n}_c(\mathbf{x}) = \mathbf{n}_c^{(\Gamma)}(\mathbf{x}), \quad \forall \mathbf{x} \in \Gamma, \end{cases} \quad (3.4)$$

where $\mathbf{n}_c^{(\Gamma)}$ is the given co-normal on the boundary curve Γ . Similarly, the mixed variational form of Willmore flow (3.2) can be written as: Find $(\mathbf{x}, \mathbf{y}) \in H^2(\mathcal{S})^3 \times H^1(\mathcal{S})^3$ such that

$$\left\{ \begin{array}{l} \int_{\mathcal{S}} \frac{\partial \mathbf{x}}{\partial t} \phi \, dA + \int_{\mathcal{S}} [\phi \otimes \mathbf{y} - \mathbf{n}(\nabla \phi)^T \nabla \mathbf{y}] \mathbf{n} \, dA \\ \quad + \int_{\mathcal{S}} 2\mathbf{n}(H^2 - K)\phi \mathbf{n}^T \mathbf{y} \, dA = \mathbf{0}, \quad \forall \phi \in H_0^1(\mathcal{S}), \\ \int_{\mathcal{S}} \mathbf{y} \psi \, dA + \frac{1}{2} \int_{\mathcal{S}} (\nabla \mathbf{x})^T \nabla \psi \, dA - \frac{1}{2} \int_{\partial \mathcal{S}} \mathbf{n}_c \psi \, ds = \mathbf{0}, \quad \forall \psi \in H^1(\mathcal{S}), \\ \mathcal{S}(0) = \mathcal{S}_0, \quad \partial \mathcal{S}(t) = \Gamma, \quad \mathbf{n}_c(\mathbf{x}) = \mathbf{n}_c^{(\Gamma)}(\mathbf{x}), \quad \forall \mathbf{x} \in \Gamma. \end{array} \right. \quad (3.5)$$

Finally, the mixed variational form of the quasi-surface diffusion flow is: Find $(\mathbf{x}, \mathbf{y}) \in H^2(\mathcal{S})^3 \times H^1(\mathcal{S})^3$ such that

$$\left\{ \begin{array}{l} \int_{\mathcal{S}} \frac{\partial \mathbf{x}}{\partial t} \phi \, dA - 2 \int_{\mathcal{S}} (\nabla \mathbf{y})^T \nabla \phi \, dA = \mathbf{0}, \quad \forall \phi \in H_0^1(\mathcal{S}), \\ \int_{\mathcal{S}} \mathbf{y} \psi \, dA + \frac{1}{2} \int_{\mathcal{S}} (\nabla \mathbf{x})^T \nabla \psi \, dA - \frac{1}{2} \int_{\partial \mathcal{S}} \mathbf{n}_c \psi \, ds = \mathbf{0}, \quad \forall \psi \in H^1(\mathcal{S}), \\ \mathcal{S}(0) = \mathcal{S}_0, \quad \partial \mathcal{S}(t) = \Gamma, \quad \mathbf{n}_c(\mathbf{x}) = \mathbf{n}_c^{(\Gamma)}(\mathbf{x}), \quad \forall \mathbf{x} \in \Gamma. \end{array} \right. \quad (3.6)$$

In the section section 3.4, systems (3.4)–(3.6) are numerically solved using mixed finite element methods.

3.2 Construction Steps of GPDE B-Spline Surfaces

Problem Description. *Given four boundary curves and the cross tangents on the curves, we need to construct a four-sided B-spline surface $\mathbf{x}(u, v) = \sum_{i=0}^{m+k-1} \sum_{j=0}^{n+k-1} \mathbf{p}_{ij} N_{i,k}(u) N_{j,k}(v)$, $(u, v) \in [0, 1]^2$ which interpolates the given boundary curves with the given tangents, and satisfies a specific GPDE in $(0, 1)^2$*

The construction steps are outlined as follows. The details of each step are given in subsequent subsections.

1. **Construct initial inner control points:** The initial inner control points can be arbitrarily given. However, to have a fast convergence of the evolution process, good initial values are necessary. We use the Coons interpolation technique [5] using the boundary control points to construct inner control points (see section 3.3).
2. **Evolve the control points:** Use a specific GPDE to evolve the inner control points, till steady state solution is achieved. The evolution includes the following steps (see section 3.4):
 - (a) Set a temporal step-size τ .

- (b) Discretize the specific GPDE in the spatial direction using a mixed finite element method (see section 3.4.1), to yield a system of nonlinear ordinary differential equations (ODEs).
- (c) Discretize the system of ODEs in the temporal direction using a semi-implicit scheme (see section 3.4.3), to yield a linear system.
- (d) Solve the linear system using an iterative approach, to yield a new approximate solution of the inner control points.
- (e) Check the termination conditions, if they are satisfied, stop the evolution, otherwise go back to step 2.

3.3 Construction of Initial Inner Control Points

Good initial inner control points yield a more efficient evolution step. We utilize the construction technique of Coons surface patch (see [5]) from the boundary curves, to calculate our initial inner control points. The construction results are

$$\mathbf{p}_{ij}^{(0)} = \mathbf{p}_{ij}^{(u)} + \mathbf{p}_{ij}^{(v)} - \mathbf{p}_{ij}^{(uv)}, \quad i = 1, \dots, m+k-2, \quad j = 1, \dots, n+k-2,$$

where (if $k = 3$),

$$\begin{aligned} \mathbf{p}_{ij}^{(u)} &= (1 - \alpha_i)\mathbf{p}_{0j} + \alpha_i\mathbf{p}_{m+k-1,j}, \\ \mathbf{p}_{ij}^{(v)} &= (1 - \beta_j)\mathbf{p}_{i0} + \beta_j\mathbf{p}_{i,n+k-1}, \\ \mathbf{p}_{ij}^{(uv)} &= (1 - \alpha_i)(1 - \beta_j)\mathbf{p}_{00} + \alpha_i(1 - \beta_j)\mathbf{p}_{m+k-1,0} \\ &\quad + (1 - \alpha_i)\beta_j\mathbf{p}_{0,n+k-1} + \alpha_i\beta_j\mathbf{p}_{m+k-1,n+k-1}, \\ \alpha_i &= \begin{cases} \frac{1}{3m}, & i = 1, \\ \frac{i-1}{m}, & i = 2, \dots, m, \\ 1 - \frac{1}{3m}, & i = m+1, \end{cases} & \beta_j = \begin{cases} \frac{1}{3n}, & j = 1, \\ \frac{j-1}{n}, & j = 2, \dots, n, \\ 1 - \frac{1}{3n}, & j = n+1. \end{cases} \end{aligned}$$

3.4 Construction of Inner Control Points

For ease of description, we reorder the basis functions and control points into one dimensional arrays. First, we order the inner basis and control points as follows:

$$\begin{aligned} \phi_{(i-1)(n+k-2)+j-1}(u, v) &= N_{i,k}(u)B_{j,k}(v), \quad i = 1, \dots, m+k-2, \quad j = 1, \dots, n+k-2, \\ \mathbf{x}_{(i-1)(n+k-2)+j-1} &= \mathbf{p}_{ij}, \quad i = 1, \dots, m+k-2, \quad j = 1, \dots, n+k-2. \end{aligned}$$

Then we order the basis and control points on the boundary. Starting with the indices $(0, 0)$, the basis functions and control points at the surface boundary are arranged in clockwise order. Using this ordering of the basis functions and control points, the spline surface patch is represented as

$$\mathbf{x}(u, v) = \sum_{j=0}^{n_0} \mathbf{x}_j \phi_j(u, v) + \sum_{j=n_0+1}^{n_1} \mathbf{x}_j \phi_j(u, v), \quad (3.7)$$

where

$$n_0 = (m + k - 2)(n + k - 2) - 1, \quad n_1 = (m + k)(n + k) - 1.$$

The mean curvature vector of the surface is represented approximately as

$$\mathbf{H}(u, v) = \sum_{j=0}^{n_1} \mathbf{h}_j \phi_j(u, v), \quad \mathbf{h}_j \in \mathbb{R}^3. \quad (3.8)$$

The coefficients \mathbf{x}_j of (3.7) are unknowns, while the coefficients \mathbf{x}_j in the second term are given. All the coefficients in (3.8) have to be calculated.

3.4.1 Spatial Discretizations

Spatial Discretizations of SDF and WF. Substituting (3.7) and (3.8) into (3.4) and (3.5), and taking the test functions ϕ as $\phi_i (i = 0, \dots, n_0)$, ψ and $\phi_i (i = 0, \dots, n_1)$, and finally noting that $\frac{\partial \mathbf{x}_j(t)}{\partial t} = \mathbf{0}$ if $j > n_0$, we obtain the following matrix representations of (3.4) and (3.5):

$$\begin{cases} M_{n_0}^{(1)} \frac{\partial X_{n_0}(t)}{\partial t} + L_{n_1}^{(1)} H_{n_1}(t) = \mathbf{0}, \\ M_{n_1}^{(2)} H_{n_1}(t) + L_{n_1}^{(2)} X_{n_1}(t) = B, \end{cases} \quad (3.9)$$

where

$$\begin{aligned} X_j(t) &= [\mathbf{x}_0^T(t), \dots, \mathbf{x}_j^T(t)]^T \in \mathbb{R}^{3(j+1)}, \\ H_{n_1}(t) &= [\mathbf{h}_0^T(t), \dots, \mathbf{h}_{n_1}^T(t)]^T \in \mathbb{R}^{3(n_1+1)}, \\ B &= [\mathbf{b}_0^T, \dots, \mathbf{b}_{n_1}^T]^T \in \mathbb{R}^{3(n_1+1)}, \\ M_{n_0}^{(1)} &= (m_{ij})_{ij=0}^{n_0, n_0}, \quad M_{n_1}^{(2)} = (m_{ij})_{ij=0}^{n_1, n_1}, \\ L_{n_1}^{(1)} &= (l_{ij}^{(1)})_{ij=0}^{n_0, n_1}, \quad L_K^{(2)} = (l_{ij}^{(2)})_{ij=0}^{n_1, K}, \end{aligned}$$

and

$$m_{ij} = \mathbf{I}_3 \int_S \phi_i \phi_j \, dA, \quad \mathbf{I}_3 \int_S \phi_i \phi_j \, dA, \quad (3.10)$$

$$\begin{aligned} l_{ij}^{(1)} &= 2 \int_S [\phi_i \otimes \phi_j - \mathbf{n}(\nabla \phi_i)^T \nabla \phi_j] \mathbf{n}^T \, dA \text{ for SDF}, \\ l_{ij}^{(1)} &= \int_S [\phi_i [\otimes \phi_j + 2\mathbf{n}(H^2 - K)\phi_j] - \mathbf{n}(\nabla \phi_i)^T \nabla \phi_j] \mathbf{n}^T \, dA \text{ for WF}, \\ l_{ij}^{(2)} &= \frac{1}{2} \mathbf{I}_3 \int_S [(\nabla \phi_i)^T \nabla \phi_j] \, dA, \\ \mathbf{b}_i &= \frac{1}{2} \int_\Gamma \mathbf{n}_c \phi_i \, ds. \end{aligned} \quad (3.11)$$

Moving the terms related to the known vertices $\mathbf{x}_{n_0+1}, \dots, \mathbf{x}_{n_1}$ in equation(3.9) to the equations's right-hand side, we rewrite (3.9) as

$$\begin{cases} M_{n_0}^{(1)} \frac{\partial X_{n_0}(t)}{\partial t} + L_{n_1}^{(1)} H_{n_1}(t) = \mathbf{0}, \\ M_{n_1}^{(2)} H_{n_1}(t) + L_{n_0}^{(2)} X_{n_0}(t) = B^{(2)}. \end{cases} \quad (3.12)$$

Note that, matrices $M_{n_0}^{(1)}$ and $M_{n_1}^{(2)}$ are symmetric and positive definite. The integrals in defining the matrix elements are computed using Gaussian quadrature formulas over rectangles. The knots and weights of the Gaussian quadrature formulas can be found in [1, 25].

Spatial Discretizations of QSDF. Substituting (3.7) and (3.8) into (3.6), taking the test functions ϕ as $\phi_i (i = 1, \dots, n_0)$ and ψ as $\phi_i (i = 1, \dots, n_1)$, and finally noticing that $\frac{\partial \mathbf{x}_j(t)}{\partial t} = \mathbf{0}$ if $j > n_0$, we obtain the following matrix form of (3.6):

$$\begin{cases} M_{n_0}^{(3)} \frac{\partial \mathcal{X}_{n_0}(t)}{\partial t} + L_{n_1}^{(3)} \mathcal{H}_{n_1}(t) = \mathbf{0}, \\ M_{n_1}^{(4)} \mathcal{H}_{n_1}(t) + L_{n_1}^{(4)} \mathcal{X}_{n_1}(t) = \mathbf{B}, \end{cases} \quad (3.13)$$

where

$$\begin{aligned} \mathcal{X}_j(t) &= [\mathbf{x}_0(t), \dots, \mathbf{x}_j(t)]^T \in \mathbb{R}^{(j+1) \times 3}, \\ \mathcal{H}_{n_1}(t) &= [\mathbf{h}_0(t), \dots, \mathbf{h}_{n_1}(t)]^T \in \mathbb{R}^{(n_1+1) \times 3}, \\ \mathbf{B} &= [\mathbf{b}_0, \dots, \mathbf{b}_{n_1}]^T \in \mathbb{R}^{(n_1+1) \times 3} \\ M_{n_0}^{(3)} &= (m_{ij})_{ij=0}^{n_0, n_0}, \quad M_{n_1}^{(4)} = (m_{ij})_{ij=0}^{n_1, n_1}, \\ L_{n_1}^{(3)} &= (l_{ij}^{(3)})_{ij=0}^{n_0, n_1}, \quad L_K^{(4)} = (l_{ij}^{(4)})_{ij=0}^{n_1, K}, \end{aligned}$$

and

$$\begin{aligned} m_{ij} &= \int_S \phi_i \phi_j \, dA, \\ l_{ij}^{(3)} &= -2 \int_S (\nabla \phi_j)^T \nabla \phi_i \, dA, \\ l_{ij}^{(4)} &= \frac{1}{2} \int_S (\nabla \phi_j)^T \nabla \phi_i \, dA, \\ \mathbf{b}_i &= \frac{1}{2} \int_\Gamma \mathbf{n}_c \phi_i \, ds. \end{aligned} \quad (3.14)$$

Moving the known terms in (3.13) to the equation's right-hand side, we obtain

$$\begin{cases} M_{n_0}^{(3)} \frac{\partial \mathcal{X}_{n_0}(t)}{\partial t} + L_{n_1}^{(3)} \mathcal{H}_{n_1}(t) = \mathbf{0}, \\ M_{n_1}^{(4)} \mathcal{H}_{n_1}(t) + L_{n_0}^{(4)} \mathcal{X}_{n_0}(t) = B^{(4)}. \end{cases} \quad (3.15)$$

Matrices $M_{n_0}^{(3)}$ $M_{n_1}^{(4)}$ are symmetric and positive definite. It should be pointed out that the size of the matrices in (3.12) are much larger than those in (3.15). But the right-handed side of (3.15) has three columns.

3.4.2 Boundary Conditions

In the boundary integrals (3.11) and (3.14), where \mathbf{n}_c is the co-normal of the surface, it is infeasible to compute these co-normals from the previous approximation, since they do not satisfy the given boundary condition. The right way is to replace \mathbf{n}_c with $\mathbf{n}_c^{(\Gamma)}$. That is

$$\mathbf{b}_i = \frac{1}{2} \int_{\Gamma} \mathbf{n}_c^{(\Gamma)} \phi_i \, ds.$$

A point to note is that the integral above does not involve the length of the tangent vector.

3.4.3 Temporal Direction Discretization

We consider only the temporal direction discretization of SDF and WF. The temporal direction discretization of QSDF is similar. Suppose we have approximate solutions $X_{n_0}^{(k)} = X_{n_0}(t_k)$ and $H_{n_1}^{(k)} = H_{n_1}(t_k)$ at $t = t_k$. We obtain approximate solutions $X_{n_0}^{(k+1)}$ and $H_{n_1}^{(k+1)}$ at $t = t_{k+1} = t_k + \tau^{(k)}$ using a semi-implicit Euler scheme. Specifically, we use the following approximation

$$\frac{X_{n_0}(t_{k+1}) - X_{n_0}(t_k)}{\tau^{(k)}} \approx \frac{\partial X_{n_0}}{\partial t}.$$

The matrices $M^{(1)}$, $M^{(2)}$, $L^{(1)}$ and $L^{(2)}$ in (3.12) are computed using the surface data at $t = t_k$. This yields a linear system with $X_{n_0}^{(k+1)}$ and $H_{n_1}^{(k+1)}$ as unknowns:

$$\begin{bmatrix} M_{n_0}^{(1)} & \tau^{(k)} L_{n_1}^{(1)} \\ L_{n_0}^{(2)} & M_{n_1}^{(2)} \end{bmatrix} \begin{bmatrix} P_{n_0}^{(k+1)} \\ H_{n_1}^{(k+1)} \end{bmatrix} = \begin{bmatrix} \tau^{(k)} B^{(1)} + M_{n_0}^{(1)} P_{n_0}^{(k)} \\ B^{(2)} \end{bmatrix}$$

Though the matrices $M^{(1)}$ and $M^{(2)}$ are symmetric and positive definite, the total matrix is neither symmetric nor positive definite. However the coefficient matrix of this system is highly sparse, hence an iterative method for its solution is desirable. We use Saad's iterative method, namely GMRES (see [16]), to solve our sparse linear system. The numerical tests show that this iterative method works very well.

4 Implementation and Experimental Results

This section presents some of our experimental results including our results on approximation errors, our results of smoothness at the surface boundaries as well as illustrative examples of surface modeling.

4.1 Numerical Test of the Convergence

Our numerical examples demonstrate that our GPDE solution is convergent. To illustrate this goal, we select some surface models which are the exact solution of certain GPDEs.

Taking $N = m + 2$, $m = n$, we compute the approximation errors of the evolved surface and the exact solution for $N = 8$. For $\tau = 0.001$, Table 4.1 lists the maximal errors of the discretized solutions and exact solution for $i = 0, 1, \dots, 10$, here i is forward steps in the temporal direction. Additionally, we compute the maximal errors of the steady solutions of the used geometric PDE and the exact solution for $N = 3, 4, \dots$. The computation results are given in Table 4.2. The initial surfaces are constructed by averaging the linear interpolations in the u and v directions. We do not use the Coons interpolation as suggested in section 3.2 to determine the initial surfaces simply because the initial surfaces constructed are too accurate for these example geometric models. To show the strength of our approach, we rather choose a poor initial surfaces.

Example 4.1 *Let \mathcal{S} be a sphere. Then $\Delta H = 0$, $H^2 - K = 0$. Hence \mathcal{S} is a steady state solution of SDF and WF. We use six four-sided spline surface patches to approximate the sphere and compute the maximal errors of our six GPDE surfaces and the exact sphere. The computation results are listed in the second and third columns of Tables 4.1 and 4.2.*

Example 4.2 *Let \mathcal{S} be a torus, which is formed by rotating a circle in the xy -plane with center $[0, R, 0]^T$ and radius r around the y -axis:*

$$\mathbf{x}(u, v) = [r \sin(u), (R + r \cos(u)) \sin(v), (R + r \cos(u)) \cos(v)]^T.$$

If $R/r = \sqrt{2}$, then \mathcal{S} is a steady state solution of WF. The fourth column of Tables 4.1 and 4.2 gives the maximal approximation errors.

Example 4.3 *Let \mathcal{S} be a cylinder. Then its mean curvature is a nonzero constant and its Gaussian curvature is zero. Hence, \mathcal{S} is a steady state solution of the SDF. The fifth column of Tables 4.1 and 4.2 gives the maximal approximation errors.*

Example 4.4 *Let \mathcal{S} be a minimal surface defined by the following equation*

$$\mathbf{x}(u, v) = [u, h(u) \cos(v), h(u) \sin(v)]^T, \tag{4.1}$$

where

$$h(u) = a \cosh(u/a + b), \quad a = 1.5, \quad b = 0.$$

Since the mean curvature of this surface is zero (but Gaussian curvature is nonzero), \mathcal{S} is a steady solution of SDF, QSDF and WF. The sixth, seventh and eighth columns of Tables 4.1 and 4.2 give the maximal approximation errors.

From the numerical results in Table 4.1 we can see that the maximal errors are monotonically decreasing as i increases. This shows that the proposed method is effective and reliable. The results in Table 4.2 further show that the approximation is very accurate.

Table 4.1: The maximal errors of evolved spline surface for $N = 8$

i	Sphere-SD	Sphere-WM	Torus-WM	Sylinder-SD	Minimal-SD	Minimal-WM	Miniml-QSD
0	0.063509	0.063509	0.120674	0.103557	0.058233	0.058233	0.058233
1	0.052931	0.052803	0.101407	0.088043	0.010384	0.010475	0.007606
2	0.034853	0.034703	0.069213	0.062165	0.003105	0.003170	0.001538
3	0.021780	0.021653	0.045283	0.041831	0.001989	0.002068	0.001182
4	0.013532	0.013439	0.029703	0.027913	0.001647	0.001736	0.001179
5	0.008420	0.008357	0.019641	0.018657	0.001486	0.001585	0.001156
6	0.005250	0.005208	0.013076	0.012501	0.001388	0.001492	0.001130
7	0.003279	0.003252	0.008752	0.008390	0.001318	0.001425	0.001105
8	0.002051	0.002035	0.005884	0.005634	0.001266	0.001375	0.001080
9	0.001286	0.001276	0.003974	0.003781	0.001226	0.001336	0.001055

Table 4.2: Maximal asymptotic errors of spline surface

N	Sphere-SD	Sphere-WM	Torus-WM	Sylinder-SD	Minimal-SD	Minimal-WM	Minimal-QSD
3	2.836e-04	2.852e-04	3.732e-03	5.906e-03	1.667e-03	1.734e-03	1.639e-03
4	3.173e-05	3.173e-05	7.197e-04	1.281e-03	1.363e-03	1.461e-03	1.566e-03
5	9.925e-05	1.037e-04	5.110e-04	5.854e-04	1.001e-03	1.141e-03	9.161e-04
6	3.524e-05	3.590e-05	2.076e-04	2.382e-04	1.360e-03	1.404e-03	8.859e-04
7	5.992e-05	6.525e-05	4.993e-04	3.637e-04	1.200e-03	1.300e-03	1.017e-03
8	3.537e-05	3.446e-05	3.001e-04	3.630e-04	9.168e-04	1.006e-03	7.773e-04
9	4.983e-05	4.944e-05	3.417e-04	2.844e-04	1.110e-03	1.244e-03	1.033e-03
10	3.998e-05	3.991e-05	2.554e-04	2.590e-04	9.843e-04	1.033e-03	9.958e-04
11	7.603e-05	7.943e-05	2.147e-04	2.571e-04	1.133e-03	1.249e-03	8.425e-04
12	7.163e-05	7.213e-05	1.939e-04	2.438e-04	1.001e-03	1.094e-03	7.688e-04

4.2 Visualization of Smoothness of Test Examples

To visually examine the constructed B-spline surfaces satisfy the specified G^1 boundary condition, we present a few example figures. In Figs 4.1—4.4, the first figures show the boundary curves and the outer surfaces. The aim of showing the outer surfaces is to present normal or tangent information. The second figures show the outer surfaces and the constructed initial surfaces. As in section 4, the initial surfaces are constructed by averaging the linear interpolations in the u and v directions. The third figures give the the outer surfaces and the GPDE B-spline surfaces with $N = 8$.

The given boundary curves and and outward tangents are computed from the exact surfaces to be approximated. In Fig 4.1, the exact surface to be approximated is a minimal surface defined by (4.1), the constructed B-spline surface is on the domain $\{[x, y]^T \in \mathbb{R}^2 : x \in [0, 1], y \in [0, 1]\}$.

In Fig 4.2, the given surface to be approximated is a cylinder defined by

$$\left(y - \frac{1}{2}\right)^2 + z^2 = \frac{1}{2}.$$

The constructed B-spline surface is corresponding to the domain $\{[x, y]^T \in \mathbb{R}^2 : x \in [0, 1], y \in [0, 1]\}$.

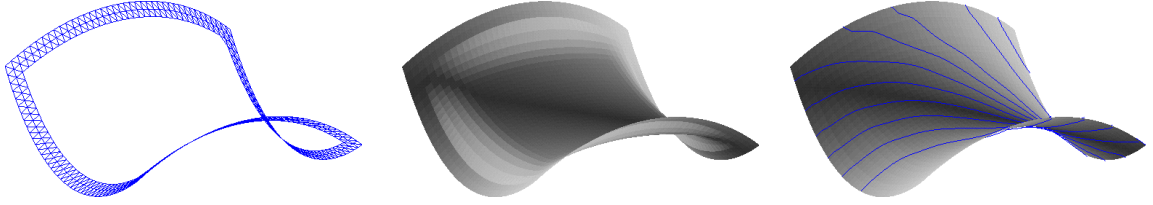


Fig 4.1: The first figure shows the input boundary curves and the outside surfaces. The second figure shows the outer surfaces and constructed initial surfaces. The third figure gives the the outer surfaces and GPDE spline surface using QSDF.

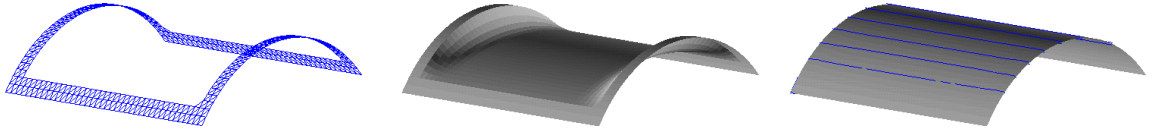


Fig 4.2: The first figure shows the input boundary curves and the outside surfaces. The second figure shows the outer surfaces and constructed initial surfaces. The third figure gives the the outer surfaces and GPDE spline surface using QSDF.

In Fig 4.3, the given surface is defined by

$$\mathbf{x}(u, v) = \left[u, v, \frac{1.25 + \cos(5.4v)}{6 + 6(3u - 1)^2} \right]^T.$$

The constructed B-spline surface is corresponding to the domain $\{[u, v]^T \in \mathbb{R}^2 : u \in [0, 1], v \in [0, 1]\}$.

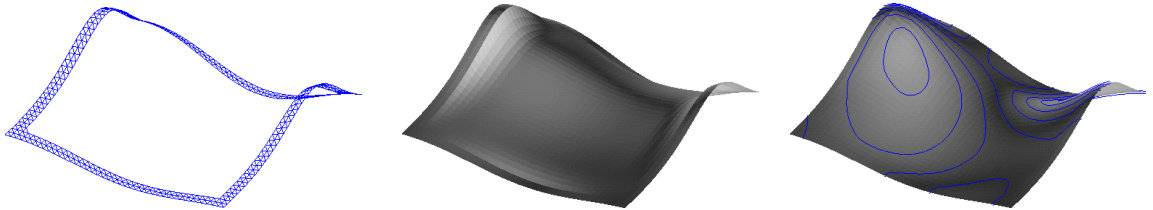


Fig 4.3: The first figure shows the input boundary curves and the outside surfaces. The second figure shows the outer surfaces and constructed initial surfaces. The third figure gives the the outer surfaces and GPDE spline surface using QSDF.

In Fig 4.4, the given surface to be approximated is

$$\mathbf{x}(u, v) = \left[u, v, e^{-\frac{81}{16}[(u-0.5)^2 + (v-0.5)^2]} \right]^T.$$

The constructed surface is corresponding to the domain $\{[u, v]^T \in \mathbb{R}^2 : u \in [0, 1], v \in [0, 1]\}$.

The surface meshes in these figures are generated by sampling uniformly the corresponding surfaces on the domain $[-2h, 1 + 2h] \times [-2h, 1 + 2h]$, where h is taken as $\frac{1}{64}$. When $[x, y]^T \in [0, 1] \times [0, 1]$, we sample on the constructed surfaces. If $[x, y]^T \notin [0, 1] \times [0, 1]$, we sample the

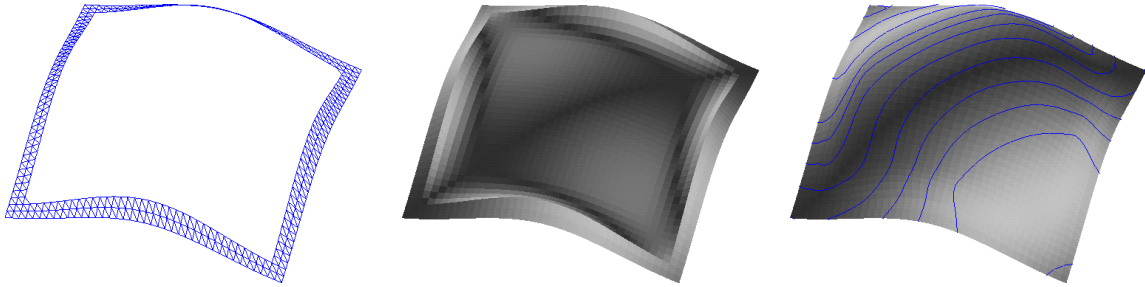


Fig 4.4: The first figure shows the input boundary curves and the out-side surfaces. The second figure shows the outer surfaces and constructed initial surfaces. The third figure gives the the outer surfaces and GPDE spline surface using QSDF.

given exact surfaces. The aim of such a sampling strategy is to show how smooth the outer surfaces join with the inner surfaces at the boundaries.

The last figures in Figs 4.1–4.4 clearly show that the constructed surfaces join the outer surfaces very smoothly. Hence G^1 condition is well satisfied.

4.3 Examples of Surface Modeling of Cloud??? Multi-Patch Object

5 Conclusions and Remarks

We have developed three different versions of the mixed finite method for solving three fourth order GPDEs. We applied these methods to solve several surface modeling problems, including surface denoising, surface blending, hole filling and mesh refinement with the G^1 continuity at the boundaries of the constructed surfaces. Our experimental results show that our numerical solution is efficient and gives very desirable results. We have also illustrated that the mixed finite element methods is more stable than our previously published divided difference method (see [24]).

References

- [1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphics, and Mathematical Tables*. America Dover Publications, Inc., 1972.
- [2] R. Bryant. A duality theorem for Willmore surfaces. *J. Diff. Geom.*, 20:23–53, 1984.
- [3] I. Chavel. *Riemannian Geometry – a Modern Introduction*. Cambridge University Press, 1993.
- [4] U. Clarenz, U. Diewald, and M. Rumpf. Anisotropic geometric diffusion in surface processing. In *Proceedings of Viz2000, IEEE Visualization*, pages 397–405, Salt Lake City, Utah, 2000.
- [5] S. A. Coons. Surfaces for computer-aided design of space forms. TR-41, Massachusetts Institute of Technology Cambridge, MA, 1967.
- [6] M. G. Cox. The numerical evaluation of B-splines. *Jour. Inst. Math. Applic.*, 10:134–149, 1972.

- [7] H. B. Curry and I. J. Schoenberg. On spline distribution and their limits: the Pólya distribution functions. *Bull. Amer. Math. Soc.*, 53:109, 1947.
- [8] C. de Boor. On calculating with B-splines. *Journal of Approximation Theory*, 6:50–62, 1972.
- [9] M. P. do Carmo. *Riemannian Geometry*. Birkhäuser, Boston, Basel, Berlin, 1992.
- [10] C. L. Epstein and M. Gage. The curve shortening flow. In A. Chorin and A. Majda, editors, *Wave Motion: Theory, Modeling, and Computation*, pages 15–59. Springer-Verlag, New York, 1987.
- [11] E. Kuwert and R. Schätzle. The Willmore flow with small initial energy. *J. Diff. Geom.*, 57(3):409–441, 2001.
- [12] E. Kuwert and R. Schätzle. Gradient flow for the Willmore functional. *Comm. Anal. Geom.*, 10(5):1228–1245, 2002.
- [13] W. W. Mullins. Theory of thermal grooving. *J. Appl. Phys.*, 28:333–339, 1957.
- [14] L. Piegl and W. Tiller. *The NURBS Book*. Springer, 1997.
- [15] L. Ramshaw. Blossoming; a connect-the-dots approach to spline. Report 19, Digital, System Research Center, Palo Alto, CA, 1987.
- [16] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM Philadelphia, PA, 2 edition, 2003.
- [17] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge, University Press, 2001.
- [18] I. J. Schoenberg. Contributions to the problem of approximation of equidistance data by analytic functions. *Quart. Appl. Math.*, 4:45–99, 1946.
- [19] L. Simon. Existence of surfaces minimizing the Willmore functional. *Commun. Analysis Geom.*, 1(2):281–326, 1993.
- [20] L. Wang and X. Xu. *Mathematical Foundations of Finite Element Methods*. Scientific Publishing Press, 2004.
- [21] G. Xu. Finite element methods for geometric modeling and processing using general fourth order geometric flows. In *GMP 2008, Lecture Notes in Computer Sciences, 4975*, pages 164–177.
- [22] G. Xu and Q. Pan. Geometric modelling by discrete surfaces patches based on geometric partial differential equations and parametric approximations. *Journal of Computer-Aided Design and Computer Graphics*, 17(12):2596–2606, 2005.
- [23] G. Xu and Q. Pan. G^1 surface modelling using fourth order geometric flows. *Computer-Aided Design*, 38(4):392–403, 2006.
- [24] G. Xu, Q. Pan, and C. Bajaj. Discrete surface modelling using partial differential equations. *Computer Aided Geometric Design*, 23(2):125–145, 2006.
- [25] G. Xu and Y. Shi. Progressive computation and numerical tables of generalized Gaussian quadrature formulas. *Journal on Numerical Methods and the Computer Application*, 27(1):9–23, 2006.
- [26] G. Xu and Q. Zhang. G^2 surface modeling using minimal mean-curvature-variation flow. *Computer-Aided Design*, 39(5):342–351, 2007.
- [27] G. Xu and Q. Zhang. A general framework for surface modeling using geometric partial differential equations. *Computer Aided Geometric Design*, 25(3):181–202, 2008.

A Derivation of Variational Forms

Now we derive the variational form formulation for the equations (3.1). Let $H^1(\mathcal{S})$ be the Sobolev space on the surface \mathcal{S} , $H_0^1(\mathcal{S})$ a subspace of $H^1(\mathcal{S})$ consisting of the functions with compact support. Using Green's formulas, the systems (3.1) and (3.2) can be rewritten as weak forms. Let $y = H$, and $\phi \in H_0^1(\mathcal{S})$ be a test function. Then we have

$$\begin{aligned}
\int_{\mathcal{S}} \mathbf{n}\phi\Delta y dA &= - \int_{\mathcal{S}} [\nabla(\mathbf{n}\phi)]^T \nabla y dA + \int_{\partial\mathcal{S}} \mathbf{n}\phi(\nabla y)^T \mathbf{n}_c ds \\
&= - \int_{\mathcal{S}} [\phi \nabla \mathbf{n} \nabla y + \mathbf{n}(\nabla \phi)^T \nabla y] dA \\
&\stackrel{(a)}{=} \int_{\mathcal{S}} [\phi \circ \mathbf{x} \nabla y - \mathbf{n}(\nabla \phi)^T \nabla y] dA \\
&= \int_{\mathcal{S}} [\phi \circ y - \mathbf{n}(\nabla \phi)^T \nabla y] dA,
\end{aligned} \tag{A.1}$$

where the validity of (a) is owing to the equality

$$\nabla \mathbf{n} + \circ \mathbf{x} = \mathbf{0}.$$

Let

$$\mathbf{y} = \mathbf{n}y = \mathbf{H}(\mathbf{x}),$$

Then it is easy to derive that

$$\nabla y = (\nabla \mathbf{y})\mathbf{n}, \quad \circ y = (\circ \mathbf{y})\mathbf{n}.$$

Substituting this into (A.1), we have

$$\int_{\mathcal{S}} \mathbf{n}\phi\Delta y dA = \int_{\mathcal{S}} [\phi \circ \mathbf{y} - \mathbf{n}(\nabla \phi)^T \nabla \mathbf{y}] \mathbf{n} dA.$$

On the other hand, for $\psi \in H^1(\mathcal{S})$, using Green's formula, we have

$$\begin{aligned}
\int_{\mathcal{S}} \mathbf{H}\psi dA &= \frac{1}{2} \int_{\mathcal{S}} \Delta \mathbf{x} \psi dA \\
&= -\frac{1}{2} \int_{\mathcal{S}} (\nabla \mathbf{x})^T \nabla \psi dA + \frac{1}{2} \int_{\partial\mathcal{S}} (\nabla \mathbf{x})^T \mathbf{n}_c \psi ds \\
&= -\frac{1}{2} \int_{\mathcal{S}} (\nabla \mathbf{x})^T \nabla \psi dA + \frac{1}{2} \int_{\partial\mathcal{S}} \mathbf{n}_c \psi ds.
\end{aligned}$$

Therefore, the mixed variational form of (3.1) is (3.4).

The derivation of the mixed variational form for (3.2) and (3.3) are similar. We omit the details.