

Modeling Deformations of General Parametric Shells Grasped by a Robot Hand

Jiang Tian and Yan-Bin Jia, *Member, IEEE*

Abstract—The robot hand applying force on a deformable object will result in a changing wrench space due to the varying shape and normal of the contact area. Design and analysis of a manipulation strategy thus depend on reliable modeling of the object's deformations as actions are performed. In this paper, shell-like objects are modeled. The classical shell theory [P. L. Gould, *Analysis of Plates and Shells*. Englewood Cliffs, NJ: Prentice-Hall, 1999; V. V. Novozhilov, *The Theory of Thin Shells*. Gronigen, The Netherlands: Noordhoff, 1959; A. S. Saada, *Elasticity: Theory and Applications*. Melbourne, FL: Krieger, 1993; S. P. Timoshenko and S. Woinowsky-Krieger, *Theory of Plates and Shells*, 2nd ed. New York: McGraw-Hill, 1959] assumes a parametrization along the two lines of curvature on the middle surface of a shell. Such a parametrization, while always existing locally, is very difficult, if not impossible, to derive for most surfaces. Generalization of the theory to an arbitrary parametric shell is therefore not immediate. This paper first extends the linear and nonlinear shell theories to describe extensional, shearing, and bending strains in terms of geometric invariants, including the principal curvatures and vectors, and their related directional and covariant derivatives. To our knowledge, this is the first nonparametric formulation of thin-shell strains. A computational procedure for the strain energy is then offered for general parametric shells. In practice, a shell deformation is conveniently represented by a subdivision surface [F. Cirak, M. Ortiz, and P. Schröder, "Subdivision surfaces: A new paradigm for thin-shell finite-element analysis," *Int. J. Numer. Methods Eng.*, vol. 47, pp. 2039–2072, 2000]. We compare the results via potential-energy minimization over a couple of benchmark problems with their analytical solutions and numerical ones generated by two commercial software packages: ABAQUS and ANSYS. Our method achieves a convergence rate that is one order of magnitude higher. Experimental validation involves regular and free-form shell-like objects of various materials that were grasped by a robot hand, with the results compared against scanned 3-D data with accuracy of 0.127 mm. Grasped objects often undergo sizable shape changes, for which a much higher modeling accuracy can be achieved using the nonlinear elasticity theory than its linear counterpart.

Index Terms—Deformable modeling, elasticity, grasping, shell.

I. INTRODUCTION

DEFORMABLE objects, which include clothes, plastic bottles, paper, magazines, ropes, wires, cables, balls, tires,

Manuscript received January 18, 2010; accepted May 6, 2010. Date of publication August 9, 2010; date of current version September 27, 2010. This paper was recommended for publication by Associate Editor S. Hirai and Editor J.-P. Laumond upon evaluation of the reviewers' comments. This work was supported in part by the Iowa State University and in part by the National Science Foundation under Grant IIS-0742334 and Grant IIS-0915876. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

The authors are with the Department of Computer Science, Iowa State University, Ames, IA 50011 USA (e-mail: jiangt@cs.iastate.edu; jia@cs.iastate.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TRO.2010.2050350

toys, sofas, fruits, vegetables, meat, processed food (e.g., cakes, dumplings, buns, and noodles), plants, animals, biological tissues, etc, are ubiquitous in our world. The ability to manipulate deformable objects is an indispensable part of the human hand's dexterity and is an important feature of intelligence.

In robotics, grasping of rigid objects has been an active research area in the past two decades [6]. The geometric foundation for form-closure, force-closure, and equilibrium grasps is now well-understood. However, grasping of deformable objects has received much less attention until recently. This is in part due to the lack of a geometric framework, and in part due to the high computational cost of modeling the physical process itself.

Since the number of degrees of freedom of a deformable object is infinite, it cannot be restrained by a finite set of contacts only. Consequently, form-closure grasp is no longer applicable. Does force-closure grasp still apply? Let us consider two fingers that squeeze a deformable object in order to grasp it. The normal at each contact point changes its direction, so does the corresponding contact-friction cone. Even if the two fingers were not initially placed at close-to-antipodal positions, the contact-friction cones may have rotated toward each other, thus resulting in a force-closure grasp.

More generally, when a robot hand applies force to grasp a soft object, deformation will result in the enlargement of the finger contact regions and the rotation of the contact normals on the object, which, in turn, will result in a change in the wrench space. Accurate and efficient modeling of the object's deformation can help us to predict the success of a grasp from its initial finger placement and applied force, and subsequently, use the prediction in the design of a grasping strategy.

This paper investigates shape modeling for shell-like objects that are grasped by a robot hand. A shell is a thin body that is bounded by two curved surfaces, whose distance, i.e., the shell thickness, is very small in comparison with the other dimensions. The locus of points, at equal distances from the two bounding surfaces, is the *middle surface* of the shell.

Shells have been studied based on the geometry of their middle surfaces that are assumed to be parametrized along the lines of curvature [19], [43], [51]. The expressions of extensional and shear strains and strain energy, although derived in a local frame at every point, are still dependent on the specific parametrization rather than on the geometric properties only. The Green-Lagrange strain tensor of a shell is presented in general curvilinear coordinates in [22] and [41]. However, the geometry of deformation is hidden due to the heavy use of covariant and contravariant tensors for strains.

The strain energy of a deformed shell depends on the geometry of its middle surface and its thickness, all prior to the deformation, as well as the displacement field. In this paper, we

will rewrite strains in terms of geometric invariants, including principal curvatures, principal vectors, and the related directional and covariant derivatives. All shell-like objects addressed in this paper satisfy the following three assumptions.

- 1) They are physically linear, but geometrically either linear or nonlinear.¹
- 2) They are considered *homogeneous* and *isotropic*, i.e., they have the same elastic properties in all directions.
- 3) Their middle surfaces are arbitrarily parametric or so approximated.

The rest of the paper is organized as follows. Section II surveys related work in the finite-element methods (FEMs) for shells and in robotics and computer graphics. Section III surveys necessary background in surface geometry. Section IV presents the displacement field on a shell that completely describes the deformation. Based on the linear elasticity theory of shells, Section V establishes that the strains and strain energy of a shell under a displacement field are decided by geometric invariants of its middle surface, including the two principal curvatures and two principal vectors. A computational procedure for arbitrary parametric shells is then described. Section VI frames the theory of nonlinear elasticity of shells in terms of geometric invariants.

Section VII sets up the subdivision-based displacement field and describes the stiffness matrix and the energy-minimization process. Section VIII compares the simulation results over two benchmark problems with their analytical solutions, and numerical ones generated by two commercial software packages ABAQUS and ANSYS. Section IX experimentally investigates the modeling of deformable objects that were grasped by a BarrettHand. It compares the linear theory for small deformations and the nonlinear theory for large deformations through validation against range data that were generated by a 3-D scanner. We will further see that the nonlinear elasticity-based modeling yields much more accurate results when large grasping forces are applied. Section X discusses modeling errors and future extensions, especially the next phase of our research on grasping of deformable objects.

This paper extends the results from our conference papers [25], [50]. It delivers a clear geometric interpretation of the shell strains. For a parametric shell, geometric invariants are now computed based on the original parametrization rather than its subdivision-surface approximation, and hence, results in more accuracy. This paper also adds simulation results for two benchmark problems, and presents more experimental data.

II. RELATED WORK

This paper is about deformable modeling, which has been studied in the elasticity theory, solid mechanics, robotics, and computer graphics across a range of applications.

¹*Physical linearity* refers to the assumption that the elongations do not exceed the limit of proportionality; therefore, the stress-strain relation is governed by Hooke's law. *Geometric nonlinearity* refers to the assumption that the angles of rotation are of a higher order than the elongations and shears. *Geometric linearity* refers to the assumption that they all are of the same order.

A. Elasticity

The FEM [4], [16], [45], for modeling deformations of a wide range of shapes, represents a body as a mesh structure, and computes the stress, strain, and displacement everywhere inside the body. FEMs are used to model the deformations of a wide range of shapes: fabric [10], a deformable object that interacts with a human hand [20], human tissue in a surgery [7], etc.

Thin-shell finite elements originated in the mid-1960s. Two comprehensive surveys [59], [60] were given by Yang *et al.* It is well-known that convergence of thin-shell elements requires C^1 interpolation, which is difficult. From a viewpoint of engineering, it is crucial to formulate models that are both physically accurate and numerically robust for arbitrary shapes.

Cirak *et al.* [9] introduced an FEM based on subdivision surfaces. By assuming linear elasticity, they presented simulation results for planar, cylindrical, and spherical shells only. The work was extended in [49] to model dynamics in textile simulation.

Other thin-shell FEMs include flat plates [61], axisymmetric shells [21], [39], and curve elements [11]. More recently, computational shell analysis in the FEM has employed techniques, including degenerated shell approach [24], stress-resultant-based formulations [1], integration techniques [5], 3-D elasticity elements [13], etc.

B. Robot Manipulation

Compared with an abundance of research in grasping of rigid objects in the past two decades, less attention has been paid to grasp the deformable objects. Work on robotic manipulation of deformable objects has been mostly limited to linear and meshed objects [34], [55]. Most of the developed models are energy-based, and some of them are not experimentally verified.

Wakamatsu *et al.* [53] examined whether force-closure and form-closure grasps can be applied to grasp deformable objects. Form-closure grasp is not applicable because deformable objects have infinite degrees of freedom, and they cannot be constrained by a finite number of contacts.

The deformation space (D-space) of an object was introduced in [18] as the C-space of all its mesh vertices, with modeling based on linear elasticity and frictionless contact. Hirai *et al.* [23] proposed a control law to grasp deformable objects, using both visual and tactile methods to control the motion of a deformable object. Most recently, a "fishbone" model, which is based on differential geometry for belt objects, was presented and experimentally verified [56].

Picking up a highly flexible linear object, such as a wire or rope, can be easily done with a vision system [42]. Knotting [32], [44], unknotting [28], and both [54] are the typical manipulation operations on this type of linear objects, which can be carried out with no need of deformable modeling.

C. Computer Graphics

Gibson and Mirtich [17] gave a comprehensive review on deformable modeling in computer graphics. The main objective in this field is to generate visual effects efficiently, rather than to

be physically accurate. Discrepancies with the theory of elasticity are tolerated, and experiments with real objects need not be conducted. For instance, the widely used formulation [47] on the surface strain energy, as the integral sum of the squares of the norms of the changes in the first and second fundamental forms, does not follow the theory of elasticity.

In this field, there are generally two approaches to model deformable objects: geometry-based approach and physics-based approach [17]. In a geometry-based approach, splines and spline surfaces, such as Bézier curves, B-splines, nonuniform rational B-splines (NURBS), are often used as representations [3], [14].

Physics-based modeling [35] of deformation takes into account the mechanics of materials and dynamics to a certain degree. Mass-spring systems, although inaccurate and slow for simulation of material with high stiffness, are extensively used in animation [8], facial modeling [48], [58], surgery [12], and simulations of cloth [2], and animals [52]. Meanwhile, the “snake model” is widely used in medical image analysis [33]. The skeleton-based method [31] achieves efficiency of deformable modeling by interpolation.

III. SOME BACKGROUND IN SURFACE GEOMETRY

Throughout the paper, we will denote by f_u for partial derivative of a function $f(u, v)$ with respect to u , and by f_{uu} the second derivative with respect to the same variable. All the vectors will be column vectors by default, and will appear in the bold face. Displacements and strain components (of all orders) will be denoted by Greek letters, as will surfaces, curvatures, and torsions by convention. In addition, points, tangents, normals, and other geometric vectors will be denoted by English letters by convention.

Let $\sigma(u, v)$ be a surface patch in 3-D. It is *regular* such that the tangent plane at every point q is spanned by the two partial derivatives σ_u and σ_v . The unit normal to the surface is $\mathbf{n} = \sigma_u \times \sigma_v / \|\sigma_u \times \sigma_v\|$. The *first fundamental form* of σ is defined as $Edu^2 + 2Fdu dv + Gdv^2$, where

$$E = \sigma_u \cdot \sigma_u, \quad F = \sigma_u \cdot \sigma_v, \quad G = \sigma_v \cdot \sigma_v \quad (1)$$

and the *second fundamental form* is defined as $Ldu^2 + 2Mdu dv + Ndv^2$, where

$$L = \sigma_{uu} \cdot \mathbf{n}, \quad M = \sigma_{uv} \cdot \mathbf{n}, \quad N = \sigma_{vv} \cdot \mathbf{n}. \quad (2)$$

A compact representation of the two fundamental forms comprises the following two symmetric matrices:

$$\mathcal{F}_I = \begin{pmatrix} E & F \\ F & G \end{pmatrix} \quad \text{and} \quad \mathcal{F}_{II} = \begin{pmatrix} L & M \\ M & N \end{pmatrix}. \quad (3)$$

The eigenvalues of $\mathcal{F}_I^{-1} \mathcal{F}_{II}$ are the two *principal curvatures* κ_1 and κ_2 at the point q . They represent the maximum and minimum rates of change in geometry when passing through q at unit speed on the patch, and are achieved in two orthogonal velocity directions, respectively, unless $\kappa_1 = \kappa_2$.² These two directions, which are denoted by unit vectors \mathbf{t}_1 and \mathbf{t}_2 , are

²The point is called *umbilic*, when $\kappa_1 = \kappa_2$. In this case, geometric variation is the same in every tangent direction.

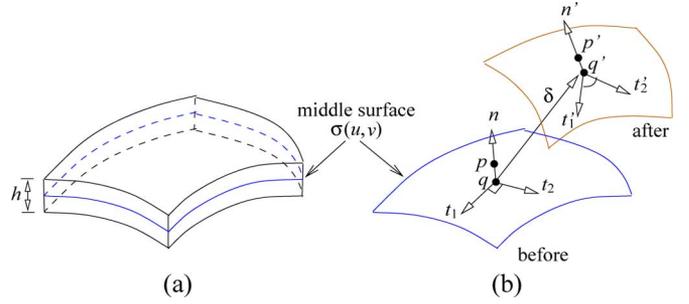


Fig. 1. Deformation of a shell. The point p in the shell is along the direction of the normal \mathbf{n} at the point q on the middle surface. After a deformation, the two points are displaced to p' and q' , respectively.

referred to as the *principal vectors*, where the indices are chosen so that $\mathbf{n} = \mathbf{t}_1 \times \mathbf{t}_2$. These three vectors define the *Darboux frame* at the point q .

The *Gaussian* and *mean curvatures* are, respectively, the determinant and half the trace of the matrix $\mathcal{F}_I^{-1} \mathcal{F}_{II}$

$$K = \kappa_1 \kappa_2 = \frac{LN - M^2}{EG - F^2} \quad (4)$$

$$H = \frac{\kappa_1 + \kappa_2}{2} = \frac{1}{2} \frac{EN - 2FM + GL}{EG - F^2}. \quad (5)$$

A curve on the patch is called a *line of curvature*, if its tangent is in a principal direction everywhere. The patch is *orthogonal*, if $F = 0$ everywhere. It is a *principal patch*, if $F = M = 0$ everywhere. In other words, a principal patch is parametrized along the two lines of curvature, one in each principal direction. On such a patch, the principal curvatures are simply $\kappa_1 = L/E$ and $\kappa_2 = N/G$, respectively, and the corresponding principal vectors are $\mathbf{t}_1 = \sigma_u / \sqrt{E}$, and $\mathbf{t}_2 = \sigma_v / \sqrt{G}$. For more details on elementary differential geometry, see [38] and [40].

IV. DISPLACEMENT FIELD ON A SHELL

Let $\sigma(u, v)$ be the middle surface of a thin shell with thickness h before the deformation, as shown in Fig. 1(a). The parametrization is regular. Every point p in the shell is along the normal direction of some point q on the middle surface, i.e., $p = q + z\mathbf{n}$, where z is the signed distance from q to p .

The displacement $\delta(u, v)$ of $q = \sigma(u, v)$ is best described in its Darboux frame as

$$\delta(u, v) = \alpha(u, v)\mathbf{t}_1 + \beta(u, v)\mathbf{t}_2 + \gamma(u, v)\mathbf{n}. \quad (6)$$

We refer to the vector field $\delta(u, v)$ as the *displacement field* of the shell. The new position of q after deformation is

$$q' = \sigma'(u, v) = \sigma(u, v) + \delta(u, v).$$

Meanwhile, the displacement of p has an additional term, which is linear in the thickness z

$$\delta(u, v) + z \begin{pmatrix} \vartheta(u, v) \\ \varphi(u, v) \\ \chi(u, v) \end{pmatrix} \quad (7)$$

from the classical shell theory [37, p. 178]. The new position p' of the point p may not be along the normal direction of q' , due

to a *transverse shear strain* that acts on the surface through \mathbf{p} and parallel to the middle surface. This type of strain tends to be much smaller than other types of strain on a shell, and it is often neglected in the classical shell theory [30], [51] under the following Kirchhoff's assumption:

Straight fibers normal to the middle surface of a shell before the deformation will, during the deformation, 1) not be elongated; and 2) remain straight and normal to the middle surface.

In this paper, we adopt the Kirchhoff's assumption and do not consider transverse shear.

When the deformation of a shell is small, the linear elasticity theory is adequate. The theory makes no distinction between the predeformation and postdeformation values of the magnitudes and positions of the areas on which the stress acts. It assumes small angles of rotation, which are of the same order of magnitude as elongations and shears. Furthermore, it neglects 1) their squares and products; and 2) neglects them as well as elongations and shears, when compared with unity and terms that do not involve these three types of terms [37, pp. 53, 83–84].

V. SMALL DEFORMATION OF A SHELL

Most of the literature [19], [37], [43], [51] on the linear elasticity theory of shells³ has assumed orthogonal curvilinear coordinates along the lines of curvature. Although in this theory, there exists a local principal patch that surrounds every point with unequal principal curvatures, most surfaces, except simple ones such as planes, cylinders, spheres, etc., do not assume such a parametrization.

The exception, to our knowledge, is [22] in which general curvilinear coordinates are used in the study of plates and shells. Nevertheless, the geometric intuition behind the kinematics of deformation is lost amidst its heavy use of covariant and contravariant tensors to express strains and stresses. The forms of these tensors still depend on a specific parametrization rather than just on the shell geometry.

Section V-A first reviews some known results on deformations and strain energy from the linear shell theory. In Section V-B, we will transform these results to make them independent of any specific parametrization, but make them rather dependent on geometric invariants, such as principal curvatures and principal vectors. In the new formulation to be derived, geometric meaning of strains will be more clearly understood. Section V-D will describe how to compute strains and strain energy on an arbitrarily parametrized shell using tools from surface geometry.

A. Strains in a Principal Patch

In this and the following section, the shell's middle surface $\sigma(u, v)$ is assumed to be a principal patch. Under a load, at a point \mathbf{q} on σ [see Fig. 1(b)], there exist *extensional strains* ϵ_1 and ϵ_2 , which are the relative increases in lengths along the two principal directions \mathbf{t}_1 and \mathbf{t}_2 , respectively. They are given as

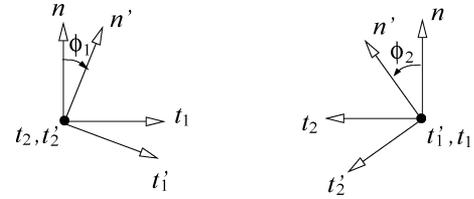


Fig. 2. Rotation of the surface normal.

follows [19, p. 219]:

$$\epsilon_1 = \frac{\alpha_u}{\sqrt{E}} + \frac{(\sqrt{E})_v}{\sqrt{EG}}\beta - \kappa_1\gamma \quad (8)$$

$$\epsilon_2 = \frac{\beta_v}{\sqrt{G}} + \frac{(\sqrt{G})_u}{\sqrt{EG}}\alpha - \kappa_2\gamma \quad (9)$$

where E, F , and G are the coefficients of the middle surface's first fundamental form that is defined in (1), and κ_1 and κ_2 are the two principal curvatures, all at \mathbf{q} .

There is also the *in-plane shear strain* ω . As shown in Fig. 1(b), \mathbf{t}'_1 and \mathbf{t}'_2 are the unit tangents from normalizing the two partial derivatives of the displaced surface σ' , respectively. These vectors are viewed as the “displaced locations” of the principal vectors \mathbf{t}_1 and \mathbf{t}_2 . The angle between \mathbf{t}'_1 and \mathbf{t}'_2 is no longer $\pi/2$, and ω is the negative change from $\pi/2$. We have $\omega = \omega_1 + \omega_2$, where [19, p. 219]

$$\omega_1 = \frac{\alpha_v}{\sqrt{G}} - \frac{(\sqrt{G})_u}{\sqrt{EG}}\beta \quad (10)$$

$$\omega_2 = \frac{\beta_u}{\sqrt{E}} - \frac{(\sqrt{E})_v}{\sqrt{EG}}\alpha. \quad (11)$$

The extensional and in-plane shear strains at \mathbf{p} , which is off the shell's middle surface, will also include some components due to the rotation of the normal \mathbf{n} . Under the assumption of small deformation, we align \mathbf{t}_2 with \mathbf{t}'_2 and view it in their common direction (see Fig. 2). Let ϕ_1 denote the amount of rotation of the normal \mathbf{n}' from \mathbf{n} about the \mathbf{t}_2 axis toward \mathbf{t}_1 . Similarly, let ϕ_2 be the amount of rotation of the normal about the \mathbf{t}_1 axis toward \mathbf{t}_2 . We have [19, pp. 209–213]

$$\phi_1 = -\frac{\gamma_u}{\sqrt{E}} - \alpha\kappa_1 \quad (12)$$

$$\phi_2 = -\frac{\gamma_v}{\sqrt{G}} - \beta\kappa_2. \quad (13)$$

It is shown that⁴ the extensional strains at $\mathbf{p} = \mathbf{q} + z\mathbf{n}$ are

$$\hat{\epsilon}_1 = \epsilon_1 + z\zeta_1 \quad (14)$$

$$\hat{\epsilon}_2 = \epsilon_2 + z\zeta_2 \quad (15)$$

³The theory is distinguished from the membrane theory that deals with elongations, but ignores shearing and bending.

⁴By dropping all terms of order $h\kappa_1$ or $h\kappa_2$, when compared with 1.

and the shearing strain at the point is

$$\hat{\omega} = \omega + z(\tau_1 + \tau_2) \quad (16)$$

where the ‘‘curvature’’ and ‘‘torsion’’ terms [19, p. 219] are

$$\zeta_1 = \frac{(\phi_1)_u}{\sqrt{E}} + \frac{(\sqrt{E})_v}{\sqrt{EG}} \phi_2 \quad (17)$$

$$\zeta_2 = \frac{(\phi_2)_v}{\sqrt{G}} + \frac{(\sqrt{G})_u}{\sqrt{EG}} \phi_1 \quad (18)$$

$$\tau_1 = \frac{(\phi_1)_v}{\sqrt{G}} - \frac{(\sqrt{G})_u}{\sqrt{EG}} \phi_2 \quad (19)$$

$$\tau_2 = \frac{(\phi_2)_u}{\sqrt{E}} - \frac{(\sqrt{E})_v}{\sqrt{EG}} \phi_1. \quad (20)$$

The geometric meanings of these terms will be revealed in Section V-B after they are rewritten into parametrization-independent forms.

Let e be the modulus of elasticity, and μ be the Poisson’s constant of the shell material. We let $\tau = \tau_1 + \tau_2$. Under Hooke’s law, the strain energy density is

$$dU_\epsilon = \frac{e}{2(1-\mu^2)} \left(\hat{\epsilon}_1^2 + 2\mu\hat{\epsilon}_1\hat{\epsilon}_2 + \hat{\epsilon}_2^2 + \frac{1-\mu}{2}\hat{\omega}^2 \right) dV.$$

The strain energy is then obtained by integration of z over the thickness interval $[-\frac{h}{2}, \frac{h}{2}]$ [37, p. 47]

$$\begin{aligned} U_\epsilon &= \int_V dU_\epsilon \\ &= \frac{e}{2(1-\mu^2)} \int_\sigma \int_{-h/2}^{h/2} \left(\hat{\epsilon}_1^2 + 2\mu\hat{\epsilon}_1\hat{\epsilon}_2 + \hat{\epsilon}_2^2 + \frac{1-\mu}{2}\hat{\omega}^2 \right) dz ds \\ &= \frac{e}{2(1-\mu^2)} \int_\sigma \left\{ h \left(\epsilon_1^2 + \epsilon_2^2 + 2\mu\epsilon_1\epsilon_2 + \frac{1-\mu}{2}\omega^2 \right) \right. \\ &\quad \left. + \frac{h^3}{12} \left(\zeta_1^2 + \zeta_2^2 + 2\mu\zeta_1\zeta_2 + \frac{1-\mu}{2}\tau^2 \right) \right\} \sqrt{EG} dudv. \end{aligned} \quad (21)$$

The linear term in h in the aforementioned equation is due to extension and shear, while the cubic term is due to bending and torsion.

B. Transformation Based on Geometric Invariants

The strains (8)–(13), (17)–(20), and the strain energy formulation (21) are applicable to the middle surface that consists of principal patches only. They need to be generalized to arbitrary parametric surfaces to widen the application scope. An important step in the generalization is to rewrite the strains in terms of geometric invariants, like principal curvatures and vectors that are independent of any specific parametrization. This is presented shortly.

The strains are given under the assumption that the middle surface $\sigma(u, v)$ is a principal patch. Let us start with rewriting

the numerator in the first term of the extensional strain (8)

$$\begin{aligned} \alpha_u &= \lim_{\Delta u \rightarrow 0} \frac{\alpha(\sigma(u + \Delta u, v)) - \alpha(\sigma(u, v))}{\Delta u} \\ &= \lim_{\Delta u \rightarrow 0} \frac{\alpha(\sigma(u, v) + \sigma_u \cdot \Delta u) - \alpha(\sigma(u, v))}{\Delta u} \\ &\stackrel{\text{def}}{=} \sigma_u[\alpha]. \end{aligned} \quad (22)$$

Here, $\sigma_u[\alpha]$ is defined to be the directional derivative of α with respect to σ_u . By the linearity of the directional-derivative operator, we rewrite the first term in (8) as follows:

$$\frac{\alpha_u}{\sqrt{E}} = \frac{\sigma_u}{\sqrt{E}}[\alpha] = \mathbf{t}_1[\alpha]. \quad (23)$$

The term $\mathbf{t}_1[\alpha]$ is independent of parametrization.

To examine the second summand in (8), we first observe that

$$\begin{aligned} \frac{(\mathbf{t}_2)_u}{\sqrt{E}} &= \lim_{\Delta u \rightarrow 0} \frac{\mathbf{t}_2(\sigma(u + \Delta u, v)) - \mathbf{t}_2(\sigma(u, v))}{\Delta u} \frac{1}{\sqrt{E}} \\ &= \lim_{\Delta u \rightarrow 0} \frac{\mathbf{t}_2(\mathbf{q} + \sigma_u \cdot \Delta u) - \mathbf{t}_2(\mathbf{q})}{\Delta u} \frac{1}{\sqrt{E}} \\ &= \lim_{\Delta u \sqrt{E} \rightarrow 0} \frac{\mathbf{t}_2(\mathbf{q} + (\sigma_u/\sqrt{E}) \cdot \Delta u \sqrt{E}) - \mathbf{t}_2(\mathbf{q})}{\Delta u \sqrt{E}} \\ &= \lim_{\Delta s \rightarrow 0} \frac{\mathbf{t}_2(\mathbf{q} + \mathbf{t}_1 \cdot \Delta s) - \mathbf{t}_2(\mathbf{q})}{\Delta s} \\ &\stackrel{\text{def}}{=} \nabla_{\mathbf{t}_1} \mathbf{t}_2. \end{aligned} \quad (24)$$

The *covariant derivative* $\nabla_{\mathbf{t}_1} \mathbf{t}_2$ measures the rate of change of the principal vector \mathbf{t}_2 as a unit-speed surface curve that passes through the point \mathbf{q} in the \mathbf{t}_1 direction. Next, we make use of the following identity:

$$(\mathbf{t}_2)_u = \frac{(\sqrt{E})_v}{\sqrt{G}} \mathbf{t}_1 \quad (25)$$

of which the proof is given in Proposition in Appendix A. Let us combine (24) and (25)

$$\frac{(\sqrt{E})_v}{\sqrt{EG}} \mathbf{t}_1 = \nabla_{\mathbf{t}_1} \mathbf{t}_2, \quad \text{and hence}$$

$$\frac{(\sqrt{E})_v}{\sqrt{EG}} = \nabla_{\mathbf{t}_1} \mathbf{t}_2 \cdot \mathbf{t}_1. \quad (26)$$

A second identity follows by symmetry

$$\frac{(\sqrt{G})_u}{\sqrt{EG}} = \nabla_{\mathbf{t}_2} \mathbf{t}_1 \cdot \mathbf{t}_2. \quad (27)$$

Substitutions of (23) and (26) into (8) results in a formulation of the extensional strain ϵ_1 , which is independent of the parametrization

$$\begin{aligned} \epsilon_1 &= \mathbf{t}_1[\alpha] + (\nabla_{\mathbf{t}_1} \mathbf{t}_2 \cdot \mathbf{t}_1) \beta - \kappa_1 \gamma \\ &= \mathbf{t}_1[\alpha] + (\nabla_{\mathbf{t}_1} \mathbf{t}_2 \cdot \mathbf{t}_1) \beta + (\nabla_{\mathbf{t}_1} \mathbf{n} \cdot \mathbf{t}_1) \gamma. \end{aligned} \quad (28)$$

The last step uses an equivalent definition of the principal curvature: $\kappa_i \stackrel{\text{def}}{=} -\nabla_{\mathbf{t}_i} \mathbf{n} \cdot \mathbf{t}_i$.

be applied to a regular patch on which the two partial derivatives are not necessarily orthogonal, i.e., $F \neq 0$. Hence, we have

$$U_\epsilon = \frac{e}{2(1-\mu^2)} \int_\sigma \left\{ h \left(\epsilon_1^2 + \epsilon_2^2 + 2\mu\epsilon_1\epsilon_2 + \frac{1-\mu}{2}\omega^2 \right) + \frac{h^3}{12} \left(\zeta_1^2 + \zeta_2^2 + 2\mu\zeta_1\zeta_2 + \frac{1-\mu}{2}\tau^2 \right) \right\} \sqrt{EG-F^2} dudv. \quad (38)$$

with all strains given in (28)–(37).

D. Strain Computation on a General Parametric Shell

With strains in terms of geometric invariants, we can compute them on an arbitrary parametric shell using tools from surface geometry. In this section, the middle surface $\sigma(u, v)$ is *not necessarily parametrized along the lines of curvature*. To compute the strains according to (28)–(37), we need to be able to evaluate the directional derivatives of the principal curvatures κ_1 and κ_2 and the displacements α, β , and γ with respect to the principal vectors \mathbf{t}_1 and \mathbf{t}_2 , as well as the covariant derivatives $\nabla_{t_i} \mathbf{t}_j$, $i, j = 1, 2$ and $i \neq j$. All these derivatives should be expressed in terms of the middle-surface parameters u and v .

1) *Differentiation of Principal Curvatures:* Let us express the principal curvatures in terms of the Gaussian curvature K and the mean curvature H (by choosing $\kappa_1 \geq \kappa_2$)

$$\kappa_1 = H + \sqrt{H^2 - K}, \quad \text{and} \quad \kappa_2 = H - \sqrt{H^2 - K}. \quad (39)$$

To obtain the partial derivatives of κ_1 and κ_2 with respect to u and v from the previous equations, we first differentiate the fundamental-form coefficients E, F, G, L, M , and N that are defined in (1) and (2). The partial derivatives of K and H are then computed using (4) and (5).

2) *Covariant Derivatives of Principal Vectors:* The principal vectors at a point \mathbf{q} are linear combinations of $\boldsymbol{\sigma}_u$ and $\boldsymbol{\sigma}_v$, which span the tangent plane

$$\mathbf{t}_1 = \xi_1 \boldsymbol{\sigma}_u + \eta_1 \boldsymbol{\sigma}_v \quad (40)$$

$$\mathbf{t}_2 = \xi_2 \boldsymbol{\sigma}_u + \eta_2 \boldsymbol{\sigma}_v. \quad (41)$$

Here, $(\xi_1, \eta_1)^T$ and $(\xi_2, \eta_2)^T$ are the eigenvectors of $\mathcal{F}_I^{-1} \mathcal{F}_{II}$ [cf., (3)] corresponding to κ_1 and κ_2 , respectively [40, p. 133]. The four coefficients, i.e., ξ_i, η_i , are derived in Appendix B.

Using (40) and (41), all the derivatives with respect to the principal vectors $\mathbf{t}_1, \mathbf{t}_2$ in (28)–(37), which are repetitive or not, can be obtained. For instance, from (22)

$$\begin{aligned} \mathbf{t}_1[\alpha] &= (\xi_1 \boldsymbol{\sigma}_u + \eta_1 \boldsymbol{\sigma}_v)[\alpha] \\ &= \xi_1 \cdot \boldsymbol{\sigma}_u[\alpha] + \eta_1 \cdot \boldsymbol{\sigma}_v[\alpha] \\ &= \xi_1 \alpha_u + \eta_1 \alpha_v. \end{aligned} \quad (42)$$

We also have, for $i, j = 1, 2$

$$\begin{aligned} \nabla_{t_i} \mathbf{t}_j &= \nabla_{\xi_i \boldsymbol{\sigma}_u + \eta_i \boldsymbol{\sigma}_v} \mathbf{t}_j \\ &= \xi_i \nabla_{\boldsymbol{\sigma}_u} \mathbf{t}_j + \eta_i \nabla_{\boldsymbol{\sigma}_v} \mathbf{t}_j \\ &= \xi_i \nabla_{\boldsymbol{\sigma}_u} (\xi_j \boldsymbol{\sigma}_u + \eta_j \boldsymbol{\sigma}_v) + \eta_i \nabla_{\boldsymbol{\sigma}_v} (\xi_j \boldsymbol{\sigma}_u + \eta_j \boldsymbol{\sigma}_v). \end{aligned} \quad (43)$$

The first summand in (43) is computed as follows:

$$\begin{aligned} \xi_i \nabla_{\boldsymbol{\sigma}_u} (\xi_j \boldsymbol{\sigma}_u + \eta_j \boldsymbol{\sigma}_v) &= \xi_i (\boldsymbol{\sigma}_u[\xi_j] \cdot \boldsymbol{\sigma}_u + \xi_j \nabla_{\boldsymbol{\sigma}_u} \boldsymbol{\sigma}_u + \boldsymbol{\sigma}_u[\eta_j] \cdot \boldsymbol{\sigma}_v + \eta_j \nabla_{\boldsymbol{\sigma}_u} \boldsymbol{\sigma}_v) \\ &= \xi_i \left(\frac{\partial \xi_j}{\partial u} \boldsymbol{\sigma}_u + \xi_j \boldsymbol{\sigma}_{uu} + \frac{\partial \eta_j}{\partial u} \boldsymbol{\sigma}_v + \eta_j \boldsymbol{\sigma}_{uv} \right). \end{aligned}$$

Here, the first step uses a fact about covariant derivatives: $\nabla_a(f\mathbf{b}) = \mathbf{a}[f] \cdot \mathbf{b} + f \cdot \nabla_a \mathbf{b}$. The second step uses (22); namely, the directional derivatives of a scalar along $\boldsymbol{\sigma}_u$ and $\boldsymbol{\sigma}_v$ are just its partial derivatives with respect to u and v , respectively. The same rule applies to the covariant derivatives of a vector with respect to $\boldsymbol{\sigma}_u$ and $\boldsymbol{\sigma}_v$. Similarly, we express the second summand in (43) in terms of partial derivatives with respect to u and v . By merging the resulting terms from the two summands, we have

$$\begin{aligned} \nabla_{t_i} \mathbf{t}_j &= \left(\xi_i \frac{\partial \xi_j}{\partial u} + \eta_i \frac{\partial \xi_j}{\partial v} \right) \boldsymbol{\sigma}_u + \left(\xi_i \frac{\partial \eta_j}{\partial u} + \eta_i \frac{\partial \eta_j}{\partial v} \right) \boldsymbol{\sigma}_v \\ &\quad + \xi_i \xi_j \boldsymbol{\sigma}_{uu} + (\xi_i \eta_j + \xi_j \eta_i) \boldsymbol{\sigma}_{uv} + \eta_i \eta_j \boldsymbol{\sigma}_{vv}. \end{aligned} \quad (44)$$

VI. LARGE DEFORMATION OF A SHELL

When a shell undergoes a large deformation, the linear elasticity theory, as presented in Section IV, is no longer adequate. This is illustrated later using the displacement caused by a rotation about the z -axis through an angle θ

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} x \\ y \\ z \end{pmatrix}.$$

No deformation happens, hence no strain occurs along the x -axis, as confirmed by the nonlinear theory [36, p. 13]

$$\epsilon_x = \frac{\partial x'}{\partial x} + \frac{1}{2} \left[\left(\frac{\partial x'}{\partial x} \right)^2 + \left(\frac{\partial y'}{\partial x} \right)^2 + \left(\frac{\partial z'}{\partial x} \right)^2 \right] = 0.$$

However, the linear elasticity theory yields a strain

$$\epsilon_x = \frac{\partial x'}{\partial x} = \cos \theta - 1$$

which is negligible only when the rotation angle θ is small.

As earlier stated, $\boldsymbol{\sigma}(u, v)$ is the middle surface of a thin shell in a regular parametrization. We look at a point $\mathbf{q} = \boldsymbol{\sigma}(u, v)$ in the middle surface with the displacement field (6) in the Darboux frame, which is defined by the two principal vectors \mathbf{t}_1 and \mathbf{t}_2 , and the normal \mathbf{n} at the point. A point $\mathbf{p} = \mathbf{q} + z\mathbf{n}$ in the shell, which projects to \mathbf{q} , has the displacement that is given as (7).

Under Kirchhoff's assumption, at \mathbf{q} , the relative elongation ϵ_{33} of a fiber along the normal \mathbf{n} , and shears ϵ_{13} and ϵ_{23} in the \mathbf{t}_1 - \mathbf{n} and \mathbf{t}_2 - \mathbf{n} planes, respectively, are zero; namely,

$$\epsilon_{33} = \epsilon_{13} = \epsilon_{23} = 0. \quad (45)$$

In the rest of section, we present the nonlinear shell theory [36, pp. 186–193], which transforms related terms into expressions in terms of geometric invariants. First, we have the relative elongations of infinitesimal line elements starting at \mathbf{q} , which,

before the deformation, were parallel to the two principal directions \mathbf{t}_1 and \mathbf{t}_2 , respectively

$$\varepsilon_{11} = \varepsilon_1 + \frac{1}{2}(\varepsilon_1^2 + \omega_1^2 + \phi_1^2) \quad (46)$$

$$\varepsilon_{22} = \varepsilon_2 + \frac{1}{2}(\varepsilon_2^2 + \omega_2^2 + \phi_2^2) \quad (47)$$

as well as the shear in the tangent plane spanned by \mathbf{t}_1 and \mathbf{t}_2

$$\varepsilon_{12} = \omega_1 + \omega_2 + \varepsilon_1\omega_2 + \varepsilon_2\omega_1 + \phi_1\phi_2. \quad (48)$$

In (46)–(48), ε_i , ω_i , and ϕ_i , $i = 1, 2$, are given in (28)–(33). Note the appearance of nonlinear terms in (46)–(48). The strains ε_{ij} , $i, j = 1, 2, 3$, which are symmetric in the indices, together constitute the Green–Lagrange strain tensor of a shell [41, pp. 201–202].

The rate of displacement, in (7), along the normal \mathbf{n} at \mathbf{q} is determined as follows:

$$\vartheta = \phi_1(1 + \varepsilon_2) - \phi_2\omega_1 \quad (49)$$

$$\varphi = \phi_2(1 + \varepsilon_1) - \phi_1\omega_2 \quad (50)$$

$$\chi = \varepsilon_1 + \varepsilon_2 + \varepsilon_1\varepsilon_2 - \omega_1\omega_2. \quad (51)$$

The relative elongations and shear at \mathbf{p} (off the middle surface) are affected by the second-order changes in geometry at its projection \mathbf{q} in the middle surface. They are characterized by six “curvature” terms, which are rewritten in terms of \mathbf{t}_1 , \mathbf{t}_2 , and \mathbf{n} , in the same way, as in Section V-B

$$\kappa_{11} = \mathbf{t}_1[\vartheta] + (\nabla_{\mathbf{t}_1} \mathbf{t}_2 \cdot \mathbf{t}_1)\varphi + (\nabla_{\mathbf{t}_1} \mathbf{n} \cdot \mathbf{t}_1)\chi$$

$$\kappa_{22} = \mathbf{t}_2[\varphi] + (\nabla_{\mathbf{t}_2} \mathbf{t}_1 \cdot \mathbf{t}_2)\vartheta + (\nabla_{\mathbf{t}_2} \mathbf{n} \cdot \mathbf{t}_2)\chi$$

$$\kappa_{12} = \mathbf{t}_1[\varphi] - (\nabla_{\mathbf{t}_1} \mathbf{t}_2 \cdot \mathbf{t}_1)\vartheta$$

$$\kappa_{21} = \mathbf{t}_2[\vartheta] - (\nabla_{\mathbf{t}_2} \mathbf{t}_1 \cdot \mathbf{t}_2)\varphi$$

$$\kappa_{13} = \mathbf{t}_1[\chi] - (\nabla_{\mathbf{t}_1} \mathbf{n} \cdot \mathbf{t}_1)\vartheta$$

$$\kappa_{23} = \mathbf{t}_2[\chi] - (\nabla_{\mathbf{t}_2} \mathbf{n} \cdot \mathbf{t}_2)\varphi.$$

Among them, κ_{11} and κ_{22} describe the changes in curvature along \mathbf{t}_1 and \mathbf{t}_2 , respectively. κ_{12} and κ_{21} together describe the twist of the middle surface in the tangent plane. κ_{13} and κ_{23} describe the twists out of the tangent plane.

The six terms κ_{ij} form the following three parameters that together characterize the variations of the curvatures of the middle surface along the principal directions:

$$\zeta_{11} = (1 + \varepsilon_1)\kappa_{11} + \omega_1\kappa_{12} - \phi_1\kappa_{13} \quad (52)$$

$$\zeta_{22} = (1 + \varepsilon_2)\kappa_{22} + \omega_2\kappa_{21} - \phi_2\kappa_{23} \quad (53)$$

$$\zeta_{12} = (1 + \varepsilon_1)\kappa_{21} + (1 + \varepsilon_2)\kappa_{12} + \omega_2\kappa_{11} + \omega_1\kappa_{22} - \phi_2\kappa_{13} - \phi_1\kappa_{23}. \quad (54)$$

Finally, we have the relative tangential elongations and shear at \mathbf{p} in terms of those at \mathbf{q} in the middle surface

$$\hat{\varepsilon}_{11} = \varepsilon_{11} + z\zeta_{11} \quad (55)$$

$$\hat{\varepsilon}_{22} = \varepsilon_{22} + z\zeta_{22} \quad (56)$$

$$\hat{\varepsilon}_{12} = \varepsilon_{12} + z\zeta_{12}. \quad (57)$$

Their derivation neglects terms in z^2 , as well as products of z with the principal curvatures $-\nabla_{\mathbf{t}_1} \mathbf{n} \cdot \mathbf{t}_1$ and $-\nabla_{\mathbf{t}_2} \mathbf{n} \cdot \mathbf{t}_2$.

In the case of a small deformation, we neglect elongations and shears, as compared with unity, for instance, $1 + \varepsilon_1 \approx 1$ in (57), as well as their products (also separately with curvature terms), such as $\varepsilon_1\omega_2$ in (48). Equations (55)–(57) then reduce to

$$\hat{\varepsilon}_{11} = \varepsilon_1 + z\kappa_{11}$$

$$\hat{\varepsilon}_{22} = \varepsilon_2 + z\kappa_{22}$$

$$\hat{\varepsilon}_{12} = \omega + z(\kappa_{12} + \kappa_{21})$$

where $\omega = \omega_1 + \omega_2$. These equations are essentially the same, as (14)–(16), in the linear elasticity theory of shells, with κ_{ii} corresponding to ζ_i , κ_{12} to τ_1 , and κ_{21} to τ_2 .

The strain energy of the shell has a similar form, as (38), in the linear case

$$U_\varepsilon = \frac{e}{2(1-\mu^2)} \int_\sigma \left\{ h \left(\varepsilon_{11}^2 + \varepsilon_{22}^2 + 2\mu\varepsilon_{11}\varepsilon_{22} + \frac{1-\mu}{2}\varepsilon_{12}^2 \right) + \frac{h^3}{12} \left(\zeta_{11}^2 + \zeta_{22}^2 + 2\mu\zeta_{11}\zeta_{22} + \frac{1-\mu}{2}\zeta_{12}^2 \right) \right\} \times \sqrt{EG - F^2} \, dudv. \quad (58)$$

VII. ENERGY MINIMIZATION OVER A SUBDIVISION-BASED DISPLACEMENT FIELD

The displacement field $\delta(u, v) = (\alpha, \beta, \gamma)^T$ of the middle surface of a shell describes its deformation completely. At the equilibrium state, the shell has minimum total potential energy [15, p. 260], which equals its strain energy (38) or (58) minus the work of applied loads. Applying calculus of variations, $\delta(u, v)$ must satisfy Euler’s (differential) equations. A variational method [57] usually approximates $\delta(u, v)$ as a linear combination of some basis functions, whose coefficients are determined via potential-energy minimization.

Since the curvature terms ζ_1 , ζ_2 , and τ , or ζ_{11} , ζ_{22} , and ζ_{12} contain second-order derivatives of the displacement, to ensure finite-bending energy, the basis functions that interpolate $\delta(u, v)$ have to be square integrable. Loop’s subdivision scheme meets this requirement [29]. Recently, the shape functions of subdivision surfaces have been used as finite-element basis functions in simulation of thin-shell deformations [9].

A subdivision surface, piecewise polynomial, is controlled by a triangular mesh with m vertices positioned at $\mathbf{x}_1, \dots, \mathbf{x}_m$ in the 3-D space. Every surface element corresponds to a triangle on the mesh, and it is determined by the locations of not only its three vertices, but also by the nine vertices in the immediate neighborhood. In Fig. 5(a), the 12 vertices that affect the shaded element are numbered with locations \mathbf{x}_i s, respectively. A point in the element is $\sum_{i=1}^{12} b_i(s, t)\mathbf{x}_i$, where s and t are barycentric coordinates that range over a unit triangle [see Fig. 5(b)]: $\{(s, t) | s \in [0, 1], t \in [0, 1 - s]\}$, and $b_i(s, t)$ are some quartic polynomials, which are called as the *box-spline basis functions* [46].

The advantage of a subdivision surface is that it can easily represent an object of arbitrary topology. The shape of a shell

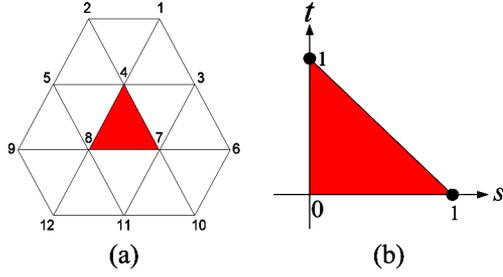


Fig. 5. (a) Regular patch with 12 control points defining a surface element that is described in (b) barycentric coordinates s and t .

after some deformation usually bears topological similarity to that before the deformation. This suggests us to approximate the deformed middle surface as a subdivision surface $\sigma'(u, v)$ over a triangular mesh that discretizes the original surface $\sigma(u, v)$.⁵ The vertices \mathbf{x}_i of $\sigma'(u, v)$ are at the positions $\mathbf{x}_i^{(0)} = \sigma(u_i, v_i)$ before the deformation; they are later displaced by $\delta_i = \mathbf{x}_i - \mathbf{x}_i^{(0)}$, respectively.

Every surface element S of σ' is parametrized with the two barycentric coordinates s and t . To compute the strain energy U_ϵ in (38) or (58), we need to set up the correspondence between (s, t) and the original parameters (u, v) . The triangular mesh of σ' induces a subdivision of the domain of the original surface, whose vertices (u_i, v_i) are the parameter values of the vertices of \mathbf{x}_i of σ' . Let $\sigma'(u_k, v_k)$ be the 12 neighboring vertices of $\sigma'(u, v)$. Then, in this domain subdivision

$$(u, v) = \sum_{k=1}^{12} b_k(s, t)(u_k, v_k). \quad (59)$$

The corresponding point on the original surface is

$$\begin{aligned} \sigma(u, v) &= \sigma\left(\sum_{i=1}^{12} b_i(s, t)(u_i, v_i)\right) \\ &\approx \sum_{i=1}^{12} b_i(s, t)\sigma(u_i, v_i) = \sum_{i=1}^{12} b_i(s, t)\mathbf{x}_i^{(0)}. \end{aligned} \quad (60)$$

Here, in the second step, the function $\sigma(u, v)$ is locally approximated as linear over the small domain region corresponding to S .

The displacement of a point on the middle surface in its Darboux frame is, as given by (6)

$$(\alpha, \beta, \gamma) = (\sigma'(u, v) - \sigma(u, v))^T(\mathbf{t}_1, \mathbf{t}_2, \mathbf{n}). \quad (61)$$

Obtaining the Jacobian with entries $\partial s/\partial u$, $\partial s/\partial v$, $\partial t/\partial u$, and $\partial t/\partial v$ from (60), the strain energy of the shell can be integrated over each subdivision element of σ' . For accuracy, all needed geometric invariants are nonetheless computed under the original parametrization σ .

If the middle surface of a shell is not parametric, but either free-form or described by an implicit equation, the subdivision

⁵Subdividing the surface domain to approximate the displacement field directly does not generate a good result, as we have found out via simulation with several surfaces, because the topology of the displacement field is unknown beforehand.

surface $\sigma'(u, v)$ for the deformed shape is then subtended by a triangular mesh over the shell's 3-D range data before the deformation. Essentially, the original middle surface is approximated by σ' , with the vertices at their predeformation positions $\mathbf{x}_i^{(0)}$.

Whether the shell is parametric or not, let m be the number of vertices of the subdivision surface σ' . The deformed shape is characterized by the column vector $\Delta = (\delta_1^T, \dots, \delta_m^T)^T$, which consists of $3m$ coordinate variables. After the deformation, the vertices are at $\mathbf{x}_i = \mathbf{x}_i^{(0)} + \delta_i$, for $1 \leq i \leq m$.

A. Stiffness Matrix

In the case of a small deformation, the system is linear, following the linear elasticity theory, and it can be easily solved. We can rewrite the strain energy U_ϵ in (38) into a matrix form

$$U_\epsilon = \Delta^T K_s \Delta \quad (62)$$

where K_s is the (symmetric) stiffness matrix, which is constructed as follows. Let us assume that there are N_e elements in the triangular control mesh of σ' . Let S_k denote the k th element. Let us number the neighboring vertices locally so that they are at $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{12}$, respectively. The displacement field $(\alpha, \beta, \gamma)^T$ of S_k is decided by $\delta_1^T, \dots, \delta_{12}^T$, where $\delta_i = (\delta_{3(i-1)+1}, \delta_{3(i-1)+2}, \delta_{3(i-1)+3})^T$, for $1 \leq i \leq 12$. Each of α, β , and γ is a linear combination of these 36 variables.

Next, we illustrate over the integral summand that involves ϵ_1^2 in (38). By its definition (28), ϵ_1 is still a linear combination of these 36 variables, say, $\epsilon_1 = \sum_{l=1}^{36} N_l \delta_l$. Let $\mathbf{t}_1 = (t_{1x}, t_{1y}, t_{1z})^T$, $\mathbf{t}_2 = (t_{2x}, t_{2y}, t_{2z})^T$, and $\mathbf{n} = (n_x, n_y, n_z)^T$. The forms of N_l s are from (28), for $1 \leq i \leq 12$, $1 \leq j \leq 3$

$$N_{3(i-1)+j} = \mathbf{t}_1 [b_i t_{1q}] + (\nabla_{t_1} \mathbf{t}_2 \cdot \mathbf{t}_1) b_i t_{2q} - \kappa_1 b_i n_q \quad (63)$$

where q is x, y , or z when $j = 1, 2, 3$, respectively, and b_i s are the subdivision basis functions. The directional and covariant derivatives in (63) are computed, according to (42) and (44), respectively. From (38), the element stiffness matrix $K^{\epsilon_1^2}$ is a 36×36 matrix (symmetric) with entries

$$K_{lp}^{\epsilon_1^2} = \frac{e}{2(1-\mu^2)} \int_{S_k} h N_l N_p d\mathcal{A}. \quad (64)$$

Similarly, we construct $K^{\epsilon_2^2}$, $K^{\epsilon_1 \epsilon_2}$, K^{ω^2} , $K^{\zeta_1^2}$, $K^{\zeta_2^2}$, $K^{\zeta_1 \zeta_2}$, and K^{τ^2} . The stiffness matrix for the element is

$$\begin{aligned} K_{S_k} &= K^{\epsilon_1^2} + K^{\epsilon_2^2} + K^{\epsilon_1 \epsilon_2} + K^{\omega^2} \\ &\quad + K^{\zeta_1^2} + K^{\zeta_2^2} + K^{\zeta_1 \zeta_2} + K^{\tau^2}. \end{aligned} \quad (65)$$

Now, we need to assemble K_{S_k} into K_s ($3m \times 3m$ matrix). The local indices of the vertices in K_{S_k} are converted to the global indices. After adding rows and columns of zeros for all vertices that are not appearing in S_k , K_{S_k} is expanded to a new $3m \times 3m$ matrix K'_{S_k} . The global stiffness matrix sums up all element contributions

$$K_s = \sum_{k=1}^{N_e} K'_{S_k}. \quad (66)$$

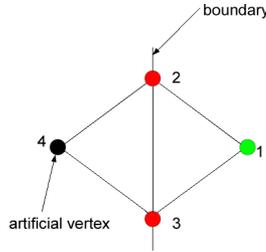


Fig. 6. Clamped boundary condition $\delta_1 = \delta_2 = \delta_3 = \delta_4 = 0$. Simply supported boundary condition $\delta_2 = \delta_3 = 0$, $\delta_4 = -\delta_1$.

B. Minimization of Potential Energy

Let $\mathbf{q}(u, v)$ denote the load field, which does work

$$U_q = \int_{\sigma} \mathbf{q}(u, v) \cdot \delta(u, v) dA = \Delta^T Q \quad (67)$$

where Q is the vector of all nodal forces. The total potential energy of a shell is

$$U = U_{\varepsilon} - U_q = \Delta^T K_s \Delta - \Delta^T Q \quad (68)$$

where the strain energy U_{ε} is given in (62).

To minimize U , a system of equations in Δ can be derived by differentiating (68), with respect to the vector, and setting all partial derivatives to zero

$$2K_s \Delta = Q. \quad (69)$$

The linear system (69) can be easily solved using Gaussian elimination or a sparse matrix method.

A large deformation is governed by the nonlinear elasticity theory. The strain energy U_{ε} in (58) no longer takes the quadratic form $\Delta^T K_s \Delta$, but rather takes a quartic form. Minimization of the total potential energy $U_{\varepsilon} - U_q$ is done iteratively. In the case of point contacts, a conical initial displacement field is placed around each contact point. Minimization over the radius of the deformed region sets the initial value of Δ . The conjugate gradient method is employed to improve on Δ , with the gradients evaluated numerically. Interpolation in the local neighborhood improves the computational efficiency. On a Dell Optiplex GX745 computer with 2.66 GHz CPU and 3.00 GB of RAM, it usually takes several minutes to obtain the solution, compared with several seconds in the linear case.

C. Boundary Conditions

Boundary conditions are handled in the same way, as described in [9]—the boundary displacements are determined only by vertices, at most, one edge away, including added artificial vertices just outside the domain. This is because of the local support within the subdivision scheme in Fig. 5. For every boundary edge, one artificial vertex is introduced. As shown in Fig. 6, vertex 4 is artificial and positioned at $\sigma_4 = \sigma_2 + \sigma_3 - \sigma_1$, where σ_1 , σ_2 , and σ_3 are the positions of the vertices 1, 2, and 3, respectively, which form a triangle. Vertex 4 affects the geometry of the surface element that corresponds to a triangle. Under the clamped condition (displacements and rotations fixed), the displacements of the vertices on the boundary and their

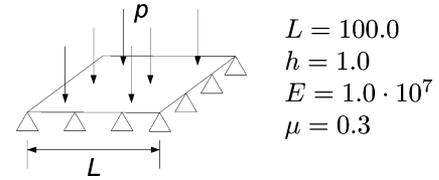


Fig. 7. Plate under gravitational load and clamped at the boundary.

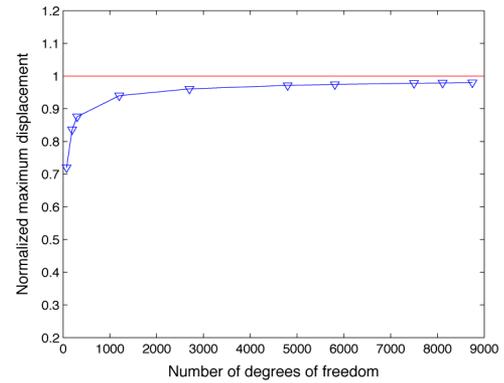


Fig. 8. Convergence of the maximum displacement for the clamped plate in Fig. 7. The number of degrees of freedom equals three times the number of vertices.

adjacent vertices, inside or outside, must be zero. Under the simply supported condition (displacements fixed and rotations free), the displacements of the vertices on the boundary must be zero, while those of the adjacent vertices, inside and outside the boundary, must be opposite to each other.

VIII. SIMULATION

The metric system is used in our simulation and experiment. For instance, the unit of Young's modulus is Pascal, while the unit of length is meter. First, simulation tests under linear elasticity are conducted on two benchmark problems, and the results are compared with their analytical solutions.⁶ These two problems in mechanics were designed to provide strict tests to deal with complex stress states.

A. Square Plate

The first benchmark problem involves a square plate under unit load of gravity ($p = 1.0$). Here, the effect of bending dominates those of elongation and shearing. As shown in Fig. 7, the plate's boundary is clamped during the deformation. Values of the plate's length L , thickness h , Young's modulus E , and Poisson's ratio μ are listed on the right side of Fig. 7.

The maximum displacement at the center of the plate is $u_{\max} \approx 0.1376$, according to the analytical solution [51, p. 202], which is in the form of a trigonometric series. Fig. 8 plots the computed maximum displacements, which is normalized over u_{\max} , against the numbers of degrees of freedom. Note that

⁶Closed-form solutions rarely exist for general thin-shell problems.

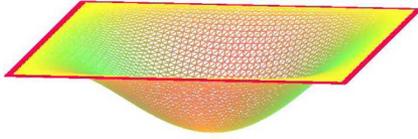


Fig. 9. Calculated deformed shape (deflection scaled) for the clamped plate in Fig. 7. Artificial vertices are marked red.

every vertex in the control mesh has three degrees of freedom. The curve plot approaches the analytical value.⁷

The geometry, load, and boundary condition are all symmetric in the example. The Young's modulus and the load represent a scaling factor only and do not affect the overall deformed shape. In Fig. 9, the load p is scaled 200 times in order to illustrate the global deformed shape. The added artificial vertices are drawn in red.

B. Comparison With Commercial Packages

Shell elements in commercial packages usually fall into two categories: degenerated 3-D solid elements and elements based on thick-shell theories, especially the Reissner–Mindlin theory [27].

A shell may be approximated as a collection of degenerated 3-D solid elements, which are simple to formulate because their strains are described in Cartesian coordinates. Meanwhile, analysis of general curved shells uses curvilinear coordinates. Although this increases the complexity of derivation, the use of curvilinear coordinates provides increased accuracy, and is thus more preferable.

The Reissner–Mindlin theory allows to shear throughout the thickness of a shell, and best models thick shells [26]. It requires C^0 interpolation only, which simplify the underlying basis functions, and is thus easy to implement. However, it often does not perform well in thin-shell analysis because of shear and membrane locking.

We will compare our method with the use of shell elements S3 and T6. The element S3 is from the commercial software ABAQUS and is based on the thick shell theory. It served as a general-purpose shell element in ABAQUS, and is widely used in industry for both thin and thick shells. The element T6 is a degenerated 3-D solid element from the SHELL93 library of another commercial package ANSYS.

Our performance criterion is accuracy in terms of the total number of degrees of freedom, which is standard in the FEM field.⁸ Here, we use a well-known benchmark problem: a cylinder with rigid-end diaphragms subjected to opposing normal point loads through its center (see Fig. 10). The radius of the cylinder is $R = 300.0$. This problem tests the ability to model deformation caused by bending and membrane stresses. The analytical solution yields a displacement of 1.8248×10^{-5} under

⁷The analytical solution considers bending only, whereas our formulation also incorporates in-plane extension, shearing and torsion, and is thus more realistic.

⁸Note that a more rigorous criterion of performance would be CPU time; however, this is quite difficult to establish because the various shell elements are run on different computer systems.

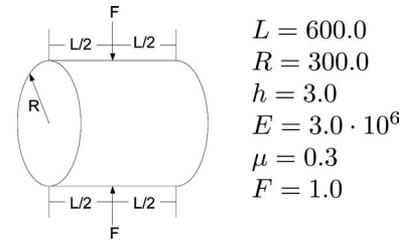


Fig. 10. Pinched cylinder.

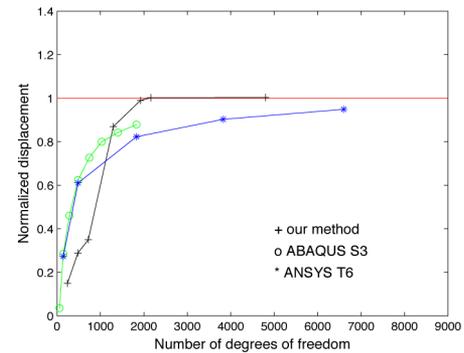


Fig. 11. Convergence of the displacement under load for the pinched cylinder in Fig. 10.

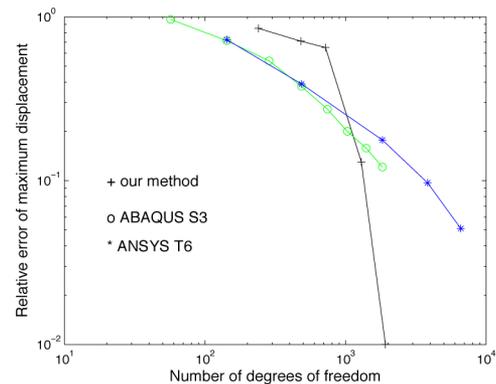


Fig. 12. Rates of convergence.

the load of $F = 1$ [41, p. 217]. The results of using elements S3 and T6 are taken from [27].

The convergence of our method to the analytical solution is shown in Fig. 11, along with those of ABAQUS and ANSYS. The vertical axis represents the deflection at the point of contact normalized over the analytical displacement value. The normalized maximum displacement converges to 1 as the number of degrees of freedom increases, which means that the solutions converge to the analytical value.

To compare the rates of convergence of the three methods, Let n denote the number of degrees of freedom in a finite-element mesh, and let r denote the relative error. The relationship between r and n is, perhaps, best illustrated by plotting $\log(r)$ against $\log(n)$. If $r = n^p$, then, $\log(r) = p \log(n)$; therefore, the relationship between $\log(r)$ and $\log(n)$ is linear with the slope p . Therefore, the rate of convergence may be conveniently measured by the slope p . As shown in Fig. 12, this slope of our

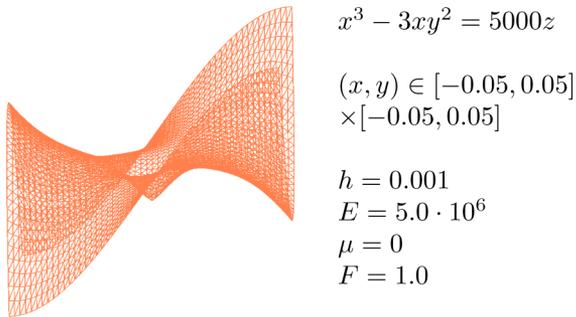


Fig. 13. Deformations of a monkey saddle. The maximum displacement under point load is 0.019 m.

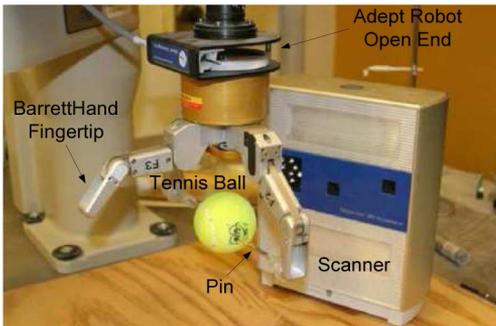


Fig. 14. Experimental setup with a tennis ball.

method is approximately -2 , which means that the relative error decays roughly at the rate of $1/n^2$. In other words, the error r decreases by a factor of 4 with every doubling of the number of degrees of freedom n . In comparison, the relative errors of both S3 and T6 decay roughly at the rate of $1/n$.

The convergence rate of our method is of an order of magnitude higher than those of ABAQUS and ANSYS.⁹ This is because the derivation of our method is based on an arbitrary shell parameterization, which is exact, and thus results in higher accuracy in implementation.

C. Algebraic Surface

Simulation test under linear elasticity is also conducted on a monkey saddle. It is worthy to note that classical shell theory does not apply directly to the shape that does not have a known parametrization along the lines of curvature. The boundary condition requires that its edge is clamped during the deformation. The result generated by our method is shown in Fig. 13. General mathematical surfaces, which are not easily modeled using the classical theory, are well in the application range of our method.

IX. EXPERIMENT

The experimental setup, as shown in Fig. 14, includes an Adept Cobra 600 manipulator, a three-fingered BarrettHand,

⁹Although both S3 and T6 converge monotonically to the reference solution, as reported in [27], T6 does so more slowly due to severe membrane locking.

and a NextEngine's desktop 3-D scanner with accuracy of 0.127 mm. Every finger of the BarrettHand has a strain gauge sensor that measures contact force. To model point contact,¹⁰ a pin is mounted on each of the two grasping fingers. A triangular mesh model of a deformed surface, due to finger contact, is generated by the scanner. We measure the modeling accuracy by matching the deformed surface from computation against the corresponding mesh model and averaging the distances from the mesh vertices to the deformed surface.¹¹

A. Tennis Ball—Linear Versus Nonlinear Elasticities

For comparison, we have conducted an experiment on a tennis ball that was grasped at antipodal positions by the BarrettHand (see Fig. 14). The rubber ball has an outer diameter of 65.0 mm and thickness of 2.5 mm. The Young's modulus of the rubber is approximated as 1 MPa, and its Poisson's ratio is approximated as 0.5. Two subdivision-based displacement fields, one for each finger contact, are used. Each field is defined over a 45 mm \times 45 mm patch, which is large enough to describe the deformed area based on our observation.

The results are given in Table I. Each row corresponds to one instance of deformation. The first column in the table lists the force exerted by each finger. The second column that consist of two subcolumns lists the deformed shapes produced by the scanner. The third and fourth columns present the corresponding deformations that are computed according to the nonlinear and linear elasticity theories, respectively.

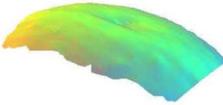
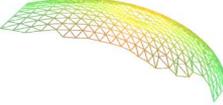
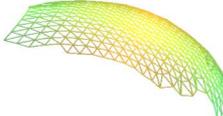
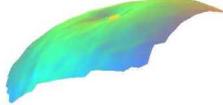
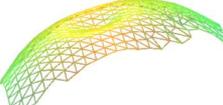
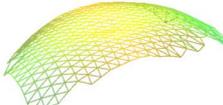
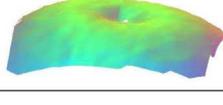
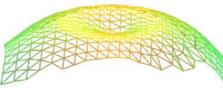
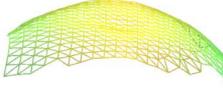
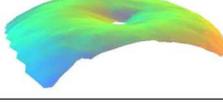
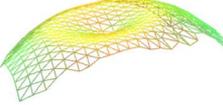
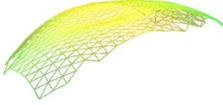
From Table I, it is shown that the nonlinear modeling results have smaller errors than the linear modeling results in three out of four grasp instances, all corresponding to large deformations. In the first instance, the two simulation results have comparable errors, which suggests that the deformation is within the range of linear elasticity. Starting from the second instance, the two methods generate shapes that are visibly different from each other. In the second instance, the shape that was generated by the nonlinear method has an obvious dent, which was comparable with the one of the real shape shown to the left, whereas the shape by the linear method, shown to the right, hardly shows any dent. We can see that the larger the force is, the bigger is the error of linear deformation. The error of nonlinear deformation does not increase with the force.

Grasping causes deformations in the regions around the contact, while the rest of the surface hardly deforms. Fig. 15 shows the deformed regions, under the finger force of 21.48 N, superposed onto the scanned undeformed model of the tennis ball. Fig. 15 corresponds to the fourth instance in Table I. The red curves, one at the top and the other at the bottom, mark the borders of these deformed regions. The measured maximum displacement of 10.27 mm is achieved at two marked points. Due to symmetry, we only display the top-deformed area. We

¹⁰Assumed between an object and a BarrettHand finger in this paper.

¹¹We select a small undeformed area on the computed surface by observation. Pick a vertex from the area, then place it at a vertex on the scanned mesh model. Align their normals, and rotate the small area to find the best match. Iterating over all vertices of the scanned mesh model will register the computed shape, after deformation onto the scanned shape.

TABLE 1
COMPARISONS BETWEEN LINEAR AND NONLINEAR DEFORMATIONS ON A TENNIS BALL

force (N)	scanned deformation		nonlinear deformation		linear deformation	
	shape	measured max disp. (mm)	shape	average error (mm)	shape	average error (mm)
10.63		2.56		0.31		0.30
16.50		6.05		0.62		0.85
20.37		9.12		0.81		2.0
21.48		10.27		0.65		2.37

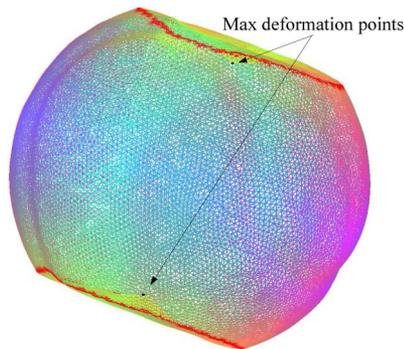


Fig. 15. Deformed tennis ball under grasping. The points in contact with the fingers have maximum displacements of 10.27 mm.

can see that the two antipodal contact points move closer under the force exerted by the two fingers. The scanned deformations on the tennis ball and the nonlinear results are within 7% of each other, from the fourth instance in Table I.

B. Rubber Duck—Free-Form Object

The surface of a real object usually has two varying principal curvatures. To demonstrate the ability to model free-form objects, we conduct an experiment on a rubber duck toy. The rubber has thickness of 2.0 mm. Its Young's modulus is approximated as 1 MPa, and Poisson's ratio is approximated as 0.5.

Fig. 16 displays the rear and the front views of the deformed rubber duck under an antipodal grasp by the BarrettHand. The average modeling error is 0.58 mm, which is within 7.4% of the scanned maximum displacement 8.56 mm.

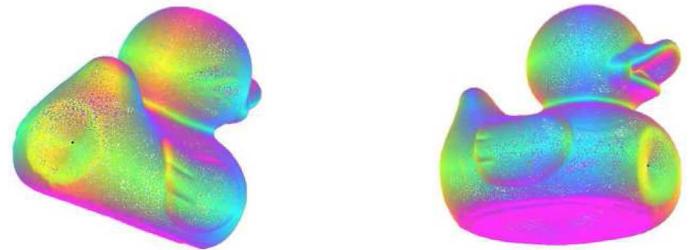


Fig. 16. Deformed rubber duck under an antipodal grasp with force of 19.22 N exerted by each finger. Two images show deformations from (left) rear view and (right) front view with maximum displacements (marked by dark points) of 8.56 and 6.73 mm, respectively.

X. DISCUSSION AND FUTURE WORK

This paper investigates deformable modeling of general shell-like objects. The first objective is to describe the linear and nonlinear shell theories, independent of a shell's middle-surface parametrization, thus making them applicable to arbitrary parametric shells, and thus to free-form shells that are well-approximated by spline or subdivision surfaces.¹² The second objective is to empirically compare our method with existing commercial software packages, which establish a convergence rate of an order of magnitude higher. The third objective is to experimentally compare the linear and nonlinear elasticity theories in the context of a deformable object that interacts with a robot hand, thus confirming that the nonlinear theory is more appropriate, given large deformations are often generated by the action of grasping.

¹²The parametric independent formulation of strains also makes it possible to treat shells described by implicit equations, although they are not common in practice.

Deformable modeling of shell-like and other objects prepares for strategies to grasp them, as already argued in Section I. Other application areas include dexterous manipulation, haptics, and computer graphics.

- 1) To dexterously manipulate a deformable object, contact force needs to be controlled based on its dynamically updated geometry under deformation.
- 2) In haptics, sensation from interacting with a deformable object is directly affected by the varying size and shape of the surface area being “touched.” Both finger-movement planning and force control (admittance or impedance) will rely on real-time updates of the local shape of contact and the global shape of the object, as well as the force distribution over the contact area.
- 3) Our modeling method, with experimental validation and the use of subdivision surfaces, is based on the physical theory of elasticity. It could potentially influence computer graphics to achieve higher realism, especially on accurate computation of strain energy and deformation under applied force.

It is worth mentioning that our invariant-based formulation is mathematically equivalent to the tensor-based one in [22]. However, this study provides much more clear geometric meanings to shell strains, which are buried in the latter formulation due to its complicated symbolism of tensor calculus.

In nonlinear modeling, an evolutionary algorithm rarely works due to its high-dimensional search space. The conjugate gradient method improves the computational efficiency with a good initial guess obtained by interpolation over the local neighborhood.

Compared with commercial packages, our method achieves a higher convergence rate. Faster convergence rate implies that a smaller number of mesh nodes are needed, which in turn results in faster running time. The invariant-based formulation of thin-shell strains increase accuracy and works with any parametrization. In contrast, commercial packages either approximate strains in Cartesian coordinates, or use thick-shell theory that could easily lead to shear and membrane locking when applied to thin shells.

There are two sources of errors in the simulation. The first is due to the discrepancy between the original surface $\sigma(u, v)$ and its “deformed” shape $\sigma'(u, v)$ as a subdivision surface under no deformation. This is because subdivision surfaces cannot represent some curved shapes exactly. The second source of error comes from modeling the deformation of the subdivision surface, a process that simplifies a variational problem, of finding a shape function that satisfies Euler’s equation, to that of determining a finite number of degrees of freedom.

In our experiment, several factors have affected the modeling accuracy: occlusion to the scanner, the scanner accuracy, and errors in the force readings (due to drifting of the zero points of the BarrettHand’s strain gauge sensors). In the tennis-ball experiment, the air pressure inside the ball also affects its deformation, but is not modeled.

In a real situation, as the object deforms, the surface region in contact with a robot finger usually grows larger and the load distribution changes. Modeling is expected to improve by con-

sidering area contacts and distributed loads. Installation of tactile array sensors on the BarrettHand can dynamically estimate contact regions on the fingertips.

We will also consider solid objects that are more common in a robot task than shell-like objects. One plan is to develop an interactive environment that can model deformations of shell-like and solid objects as the shape changes. Such an interface will facilitate future analysis and synthesis of grasp strategies for these types of objects.

For grasp analysis and synthesis, we will begin with two-fingered squeeze grasps of deformable objects. We intend to characterize the evolution of contact-friction cones, design grasp-synthesis algorithms under the energy principles, examine the roles of elasticity constants, and look into issues such as grasp stability and slip detection.

APPENDIX A

Proposition 1: The following equations hold for partial derivatives of the principal vectors \mathbf{t}_1 and \mathbf{t}_2 on a principal patch $\sigma(u, v)$:

$$(\mathbf{t}_1)_v = \frac{(\sqrt{G})_u}{\sqrt{E}} \mathbf{t}_2 \quad (70)$$

$$(\mathbf{t}_2)_u = \frac{(\sqrt{E})_v}{\sqrt{G}} \mathbf{t}_1. \quad (71)$$

Proof: Due to symmetry, we only need to prove one equation, say (71). Let us express the derivative $(\mathbf{t}_2)_u$ in the Darboux frame, which is defined by \mathbf{t}_1 , \mathbf{t}_2 , and \mathbf{n} . Differentiating the equation $\mathbf{t}_2 \cdot \mathbf{t}_2 = 1$, with respect to u , immediately yields $(\mathbf{t}_2)_u \cdot \mathbf{t}_2 = 0$. Next, we differentiate $\mathbf{t}_2 \cdot \mathbf{n} = 0$ with respect to u

$$(\mathbf{t}_2)_u \cdot \mathbf{n} + \mathbf{t}_2 \cdot \mathbf{n}_u = 0.$$

Here, \mathbf{n}_u is the derivative of \mathbf{n} along the principal direction $\mathbf{t}_1 = \sigma_u / \|\sigma_u\|$, and hence must be a multiple of \mathbf{t}_1 .¹³ Therefore, the previous equation implies $(\mathbf{t}_2)_u \cdot \mathbf{n} = 0$.

Thus, $(\mathbf{t}_2)_u$ has no component along \mathbf{t}_2 or \mathbf{n} . We only need to determine its projection onto \mathbf{t}_1 . First, differentiate $\sigma_u \cdot \sigma_v = 0$ with respect to u , to obtain

$$\sigma_{uu} \cdot \sigma_v = -\sigma_u \cdot \sigma_{uv}. \quad (72)$$

Next, we differentiate $\mathbf{t}_2 \cdot \mathbf{t}_1 = 0$ with respect to u

$$\begin{aligned} (\mathbf{t}_2)_u \cdot \mathbf{t}_1 &= -\mathbf{t}_2 (\mathbf{t}_1)_u = -\mathbf{t}_2 \left(\frac{\sigma_u}{\sqrt{E}} \right)_u \\ &= -\mathbf{t}_2 \left(\frac{\sigma_{uu}}{\sqrt{E}} + \left(\frac{1}{\sqrt{E}} \right)_u \sigma_u \right) \\ &= -\mathbf{t}_2 \frac{\sigma_{uu}}{\sqrt{E}} = -\frac{\sigma_v \cdot \sigma_{uu}}{\sqrt{EG}} \\ &= \frac{1}{\sqrt{G}} \frac{\sigma_u \cdot \sigma_{uv}}{\sqrt{E}} \quad \text{by (72)} \\ &= \frac{(\sqrt{E})_v}{\sqrt{G}}, \quad \text{since } E = \sigma_u \cdot \sigma_u. \quad \blacksquare \end{aligned}$$

¹³One can show that $\mathbf{n}_u = -E\kappa_1 \mathbf{t}_1$, although the details are omitted.

APPENDIX B

We derive the four coefficients $\xi_1, \eta_1, \xi_2,$ and η_2 in (40) and (41), as well as their partial derivatives with respect to u and v . Since the principal curvatures $\kappa_i, i = 1, 2,$ are eigenvalues of the matrix $\mathcal{F}_I^{-1}\mathcal{F}_{II}$, we have

$$\begin{aligned} 0 &= \det(\mathcal{F}_{II} - \kappa_i \mathcal{F}_I) \\ &= (L - \kappa_i E)(N - \kappa_i G) - (M - \kappa_i F)^2. \end{aligned} \quad (73)$$

There are two cases: 1) $L - \kappa_i E = N - \kappa_i G = 0,$ for $i = 1$ or $2;$ and 2) either $L - \kappa_i E \neq 0$ or $N - \kappa_i G \neq 0,$ for both $i = 1$ and $i = 2.$

In case 1), $M - \kappa_i F = 0$ by (73). Therefore, $\mathcal{F}_{II} - \kappa_i \mathcal{F}_I = 0,$ i.e.,

$$\mathcal{F}_I^{-1}\mathcal{F}_{II} = \kappa_i I_2$$

where I_2 is the 2×2 identity matrix. The two eigenvalues of $\mathcal{F}_I^{-1}\mathcal{F}_{II}$, namely, κ_1 and $\kappa_2,$ must be equal. Any tangent vector is a principal vector. We let

$$\mathbf{t}_1 = \frac{\boldsymbol{\sigma}_u}{\sqrt{E}}, \quad \text{with} \quad \begin{pmatrix} \xi_1 \\ \eta_1 \end{pmatrix} = \begin{pmatrix} 1 \\ \sqrt{E} \\ 0 \end{pmatrix} \quad \text{by (40).}$$

The other principal vector $\mathbf{t}_2 = \xi_2 \boldsymbol{\sigma}_v + \eta_2 \boldsymbol{\sigma}_v$ is orthogonal to $\mathbf{t}_1.$ Therefore

$$(\xi_2 \boldsymbol{\sigma}_u + \eta_2 \boldsymbol{\sigma}_v) \cdot \boldsymbol{\sigma}_u = 0, \quad \text{i.e.,} \quad \xi_2 E + \eta_2 F = 0. \quad (74)$$

To determine ξ_2 and $\eta_2,$ we need to use one more constraint, i.e., $\mathbf{t}_2 \cdot \mathbf{t}_2 = 1,$ which is rewritten as follows:

$$E\xi_2^2 + 2F\xi_2\eta_2 + G\eta_2^2 = 1. \quad (75)$$

Substituting (74) into (75) yields

$$\xi_2 = \mp \sqrt{\frac{F^2}{E(EG - F^2)}}, \quad \text{and} \quad \eta_2 = \pm \sqrt{\frac{E}{EG - F^2}}. \quad (76)$$

In case 2), $L - \kappa_i E \neq 0$ or $N - \kappa_i G \neq 0,$ for both $i = 1, 2.$ For $i = 1, 2,$ we know that

$$(\mathcal{F}_{II} - \kappa_i \mathcal{F}_I) \begin{pmatrix} \xi_i \\ \eta_i \end{pmatrix} = 0. \quad (77)$$

Equation (77) expands into four scalar equations according to (3)

$$(L - \kappa_i E)\xi_i + (M - \kappa_i F)\eta_i = 0 \quad (78)$$

$$(M - \kappa_i F)\xi_i + (N - \kappa_i G)\eta_i = 0. \quad (79)$$

For each i value, three subcases arise, which are as follows.

- $L - \kappa_i E = 0,$ but $N - \kappa_i G \neq 0.$ It follows from (73) that $M - \kappa_i F = 0.$ Thus, (79) gives us $\eta_i = 0.$ ξ_i has an exponent 2, i.e., $\mathbf{t}_i \cdot \mathbf{t}_i = E\xi_i^2 = 1,$ we can obtain $\xi_i = \pm \frac{1}{\sqrt{E}}.$
- $L - \kappa_i E \neq 0,$ but $N - \kappa_i G = 0.$ This is the symmetric case of a). The coefficients are

$$\begin{pmatrix} \xi_i \\ \eta_i \end{pmatrix} = \begin{pmatrix} 0 \\ \pm \frac{1}{\sqrt{G}} \end{pmatrix}.$$

- $L - \kappa_i E \neq 0,$ and $N - \kappa_i G \neq 0.$ From (78), we have

$$\xi_i = -\frac{M - \kappa_i F}{L - \kappa_i E} \eta_i. \quad (80)$$

Substitution of (80) into (75) yields a quadratic equation with the solution

$$\eta_i = \pm \sqrt{\frac{L - \kappa_i E}{EN - 2FM + LG - 2\kappa_i(EG - F^2)}}. \quad (81)$$

In all the expressions of ξ_i and $\eta_i,$ the signs are chosen such that $\mathbf{t}_1 \times \mathbf{t}_2 = \mathbf{n}.$

The gradients $\nabla \xi_i = (\frac{\partial \xi_i}{\partial u}, \frac{\partial \xi_i}{\partial v})$ and $\nabla \eta_i = (\frac{\partial \eta_i}{\partial u}, \frac{\partial \eta_i}{\partial v}), i = 1, 2,$ are obtained by differentiation of appropriate forms of ξ_i and η_i that hold for all points in some neighborhood, which are not necessarily the ones at the point.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their careful readings and valuable feedback. They would also like to acknowledge the anonymous reviews of the related conference papers [25], [50].

REFERENCES

- J. H. Argyris and D. W. Scharpf, "The SHEBA family of shell elements for the matrix displacement method. Part I. Natural definition of geometry and strains," *Aeronaut. J. Roy. Aeronaut. Soc.*, vol. 72, pp. 873–878, 1968.
- D. Baraff and A. Witkin, "Large steps in cloth simulation," in *Proc. ACM SIGGRAPH*, 1998, pp. 43–54.
- R. Bartels, J. Beatty, and B. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*. Los Altos, CA: Morgan Kaufmann, 1987.
- K. J. Bathe, *Finite Element Procedures*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- T. Belytschko and C. Tsay, "A stabilization procedure for the quadrilateral plate element with one-point quadrature," *Int. J. Numer. Methods Eng.*, vol. 19, pp. 405–419, 1983.
- A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, pp. 348–353.
- M. Bro-Nielsen and S. Cotin, "Real-time volumetric deformable models for surgery simulation using finite elements and condensatoin," in *Proc. Eurograph*, 1996, pp. 57–66.
- J. Chadwick, D. Haumann, and R. Parent, "Layered construction for deformable animated characters," in *ACM SIGGRAPH*, 1989, pp. 243–252.
- F. Cirak, M. Ortiz, and P. Schröder, "Subdivision surfaces: A new paradigm for thin-shell finite-element analysis," *Int. J. Numer. Methods Eng.*, vol. 47, pp. 2039–2072, 2000.
- J. Collier, B. Collier, G. O'Toole, and S. Sargand, "Drape prediction by means of finite-element analysis," *J. Textile Inst.*, vol. 82, pp. 96–107, 1991.
- J. J. Connor and C. A. Brebbia, "A stiffness matrix for a shallow rectangular shell element," *J. Eng. Mech. Div.*, vol. 93, pp. 43–65, 1967.
- F. Conti, O. Khatib, and C. Baur, "Interactive rendering of deformable objects based on a filling sphere modeling approach," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, pp. 3716–3721.
- A. Dorfmann and R. B. Nelson, "Three-dimensional finite element for analyzing thin plate/shell structures," *Int. J. Numer. Methods Eng.*, vol. 38, pp. 3453–3482, 1995.
- G. Farin, *Curves and Surfaces for Computer Aided Geometric Design*, 2nd ed. New York: Academic, 1990.
- R. T. Fenner, *Engineering Elasticity: Application of Numerical and Analytical Techniques*. New York: Ellis Horwood, 1986.
- R. H. Gallagher, *Finite Element Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

- [17] S. F. Gibson and B. Mirtich, "A survey of deformable models in computer graphics," Mitsubishi Electric Res. Labs., Cambridge, MA, Tech. Rep. TR-97-10, 1997.
- [18] K. Gopalakrishnan and K. Goldberg, "D-space and deform closure grasps of deformable parts," *Int. J. Robot. Res.*, vol. 24, pp. 899–910, 2005.
- [19] P. L. Gould, *Analysis of Plates and Shells*. Englewood Cliffs, NJ: Prentice-Hall, 1999.
- [20] J. P. Gourret, N. Magnenat-Thalmann, and D. Thalmann, "Simulation of object and human skin deformations in a grasping task," in *Proc. ACM SIGGRAPH*, 1989, pp. 21–30.
- [21] P. E. Grafton and D. R. Strome, "Analysis of axisymmetric shells by the direct stiffness method," *J. AIAA*, vol. 3, pp. 2138–2145, 1963.
- [22] A. E. Green and W. Zerna, *Theoretical Elasticity*. Oxford: Clarendon, 1968.
- [23] S. Hirai, T. Tsuboi, and T. Wada, "Robust grasping manipulation of deformable objects," in *Proc. IEEE Symp. Assembly Task Plann.*, 2001, pp. 411–416.
- [24] T. J. R. Hughes and W. K. Liu, "Nonlinear finite element analysis of shells: Part I. Three-dimensional shells," *Comput. Methods Appl. Mech. Eng.*, vol. 26, pp. 331–362, 1981.
- [25] Y.-B. Jia and J. Tian, "Deformations of general parametric shells: Computation and experiment," in *Proc. IEEE/RSJ Intl. Conf. Intell. Robots Syst.*, 2008, pp. 1796–1803.
- [26] A. Kamoulakos, "Understanding and improving the reduced integration of Mindlin shell elements," *Int. J. Numer. Methods Eng.*, vol. 26, pp. 2009–2029, 1988.
- [27] A. Laulusa, O. A. Bauchau, J.-Y. Choi, V. B. C. Tan, and L. Li, "Evaluation of some shear deformable shell elements," *Int. J. Solids Struct.*, vol. 43, pp. 5033–5054, 2006.
- [28] A. M. Ladd and L. E. Kavraki, "Using motion planning for knot untangling," *Int. J. Robot. Res.*, vol. 23, pp. 797–808, 2004.
- [29] C. Loop, "Smooth subdivision surfaces based on triangles," M.S. thesis, Univ. Utah, Salt Lake City, UT, 1987.
- [30] A. E. H. Love, *A Treatise on the Mathematical Theory of Elasticity*, 4th ed. New York: Dover, 1927.
- [31] Q. Luo and J. Xiao, "Contact and deformation modeling for interactive environments," *IEEE Trans. Robot.*, vol. 23, no. 3, pp. 416–430, Jun. 2007.
- [32] T. Matsuno and T. Fukuda, "Manipulation of flexible rope using topological model based on sensor information," in *Proc. IEEE/RSJ Intl. Conf. Intell. Robots Syst.*, 2006, pp. 2638–2643.
- [33] T. McInerney and D. Terzopoulos, "Deformable models in medical image analysis: A survey," *Med. Image Anal.*, vol. 1, pp. 91–108, 1996.
- [34] M. Moll and L. E. Kavraki, "Path planning for deformable linear objects," *IEEE Trans. Robot. Automat.*, vol. 22, no. 4, pp. 625–636, Aug. 2006.
- [35] A. Nealen, M. Müller, R. Keiser, E. Boxerman, and M. Carlson, "Physically based deformable models in computer graphics," *Comput. Graph. Forum*, vol. 25, pp. 809–836, 2006.
- [36] V. V. Novozhilov, *Foundations of the Nonlinear Theory of Elasticity*. Rochester, NY: Graylock, 1953; Dover, 1999.
- [37] V. V. Novozhilov, *The Theory of Thin Shells*. Gronigen, The Netherlands: Noordhoff, 1959.
- [38] B. O'Neill, *Elementary Differential Geometry*. New York: Academic, 1966.
- [39] E. P. Popov, J. Penzien, and Z. A. Lu, "Finite element solution for axisymmetric shells," *J. Eng. Mech. Div.*, vol. 90, pp. 119–145, 1965.
- [40] A. Pressley, *Elementary Differential Geometry*. New York: Springer-Verlag, 2001.
- [41] J. N. Reddy, *An Introduction to Nonlinear Finite Element Analysis*. London, U.K.: Oxford Univ. Press, 2004.
- [42] A. Remde, D. Henrich, and H. Worn, "Picking-up deformable linear objects with industrial robots," presented at the Int. Symp. Robot., Tokyo, Japan, 1999.
- [43] A. S. Saada, *Elasticity: Theory and Applications*. Melbourne, FL: Krieger, 1993.
- [44] M. Saha and P. Isto, "Motion planning for robotic manipulation of deformable linear objects," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 2478–2484.
- [45] L. Segerlind, *Applied Finite Element Analysis*. New York: Wiley, 1984.
- [46] J. Stam, "Evaluation of loop subdivision surfaces," in *Proc. ACM SIGGRAPH [CD-ROM]*, 1998.
- [47] D. Terzopoulos, J. C. Platt, and A. H. Barr, "Elastically deformable models," in *Proc. ACM SIGGRAPH*, 1987, pp. 205–214.
- [48] D. Terzopoulos and K. Waters, "Physically-based facial modeling, analysis, and animation," *J. Vis. Comput. Anim.*, vol. 1, pp. 73–80, 1990.
- [49] B. Thomaszewski, M. Wacker, and W. Straßer, "A consistent bending model for cloth simulation with corotational subdivision finite elements," in *Proc. Eurograph/SIGGRAPH Symp. Comp. Anim.*, 2006, pp. 107–116.
- [50] J. Tian and Y.-B. Jia, "Modeling deformable shell-like objects grasped by a robot hand," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 1297–1302.
- [51] S. P. Timoshenko and S. Woinowsky-Krieger, *Theory of Plates and Shells*, 2nd ed. New York: McGraw-Hill, 1959.
- [52] X. Tu and D. Terzopoulos, "Artificial fishes: Physics, locomotion, perception, behavior," in *Proc. ACM SIGGRAPH*, 1994, pp. 43–50.
- [53] H. Wakamatsu, S. Hirai, and K. Iwata, "Static analysis of deformable object grasping based on bounded force closure," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 1996, pp. 3324–3329.
- [54] H. Wakamatsu, E. Arai, and S. Hirai, "Knotting/unknotted manipulation of deformable linear objects," *Int. J. Robot. Res.*, vol. 25, pp. 371–395, 2006.
- [55] H. Wakamatsu and S. Hirai, "Static modeling of linear object deformation based on differential geometry," *Int. J. Robot. Res.*, vol. 23, pp. 293–311, 2004.
- [56] H. Wakamatsu, E. Morinaga, E. Arai, and S. Hirai, "Deformation modeling of belt object with angles," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 606–611.
- [57] K. Washizu, *Variational Methods in Elasticity and Plasticity*. New York: Pergamon, 1968.
- [58] K. Waters, "A muscle model for animating three-dimensional facial expression," in *Proc. ACM SIGGRAPH*, 1987, pp. 17–24.
- [59] H. T. Y. Yang, S. Saigal, and D. G. Liaw, "Advances of thin shell finite elements and some applications—Version 1," *Comput. Struct.*, vol. 35, pp. 481–504, 1990.
- [60] H. T. Y. Yang, S. Saigal, A. Masud, and R. K. Kapania, "A survey of recent shell finite elements," *Int. J. Numer. Methods Eng.*, vol. 47, pp. 101–127, 2000.
- [61] O. C. Zienkiewicz, *The Finite Element Method*, 3rd ed. New York: McGraw-Hill, 1979.



Jiang Tian received the B.S. degree in automatic control from Harbin Engineering University, Harbin, China, in 2001, the M.S. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China, in 2004, and the M.S. and Ph.D. degrees in computer science from Iowa State University, Ames, in 2007 and 2010, respectively.

His current research interests include modeling and grasping of deformable objects, physics-based computer graphics, shape reconstruction from tactile data, and robotics.

Dr. Tian has been a Reviewer for the IEEE International Conferences on Robotics and Automation, the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), and the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. He was a Session Co-Chair at IROS 2008.



Yan-Bin Jia (S'92–M'98) received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 1988 and the M.S. and Ph.D. degrees in robotics from Carnegie Mellon University, Pittsburgh, PA, in 1993 and 1997, respectively.

He is currently an Associate Professor with the Department of Computer Science, Iowa State University, Ames. He was a Co-Editor of a Special Issue (on Tactile Presence) of the *International Journal of Robotics Research* in 2000. His research interests

include modeling and grasping of deformable objects, simultaneous impacts, billiard robots, and shape localization, recognition, and reconstruction from tactile data.

Dr. Jia is currently an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. He also served on the Conference Editorial Board of the IEEE Robotics and Automation Society from 2006 to 2008. He was the recipient of the National Science Foundation Faculty Early Career Development Award in 2002. One of his papers was a Semi-Finalist for the Best Conference Paper Award, and one was a Finalist for the Anton Philips Best Student Paper Award at the IEEE International Conferences on Robotics and Automation in 1996 and 1998, respectively.