Contact and Deformation Modeling for Interactive Environments

Qi Luo, *Member, IEEE*, and Jing Xiao, *Senior Member, IEEE*

***Abstract*—Contact and deformation modeling for interactive environments has seen many applications, from surgical simulation and training, to virtual prototyping, to teleoperation, etc., where both visual feedback and haptic feedback are needed. High-quality feedback demands a high level of physical realism as well as a high update rate in rendering, which are often conflicting requirements. In this paper, we present a unique approach to modeling force and deformation between a rigid body and an elastic object under complex contacts, which achieves a good compromise of reasonable physical realism and real-time update rate (at least 1 kHz). We simulate contact forces based on a nonlinear physical model. We further introduce a novel approximation of material deformation suitable for interactive environments based on applying Bernoulli–Euler bending beam theory to the simulation of elastic shape deformation. Our approach is able to simulate the contact forces exerted upon the rigid body (that can be virtually held by a user via a haptic device) not only when it forms one or more than one contact with the elastic object, but also when it moves compliantly on the surface of the elastic object, taking friciton into account. Our approach is also able to simulate the global and local shape deformation of the elastic object due to contact. All the simulations can be performed in a combined update rate of over 1 kHz, which we demonstrate in several examples.**

***Index Terms*—Bending beam theory, compliant motion, contact modeling, deformable object modeling, haptic rendering, interactive environment, multiple contacts, nonhomogeneous material.**

I. Introduction

MODELING deformable objects in contact has been studied both for graphics rendering and for haptic rendering. Gibson and Mirtich [1] provided a very detailed and complete survey on deformable modeling used in graphics rendering. More recent surveys on graphics and haptic rendering involving deformable objects can be found in [2] and [3]. Existing work can be divided mainly into two large categories of approaches: purely geometric approaches (including methods based on splines and patches and free-form deformation methods), and physically based approaches (based on mass-spring models and continuum models).

Manuscript received April 30, 2004; revised September 25, 2006. This paper was recommended for publication by Associate Editor D. Prattichizzo and Editor H. Arai upon evaluation of the reviewers' comments. This work was supported in part by the National Science Foundation under Grants EIA-0224423 and IIS-0328782. This paper was presented in part at the Robotics: Science and Systems Conference, Cambridge, MA, June 2005.

The authors are with the Department of Computer Science, College of Computing and Informatics, University of North Carolina-Charlotte, Charlotte, NC 28223 USA (e-mail: xiao@uncc.edu; qluo@uncc.edu).

Color versions of Figs. 2–6, 12–18, and 10–13 are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2007.895058

While graphics rendering only needs to make the modeled object deformation *look* realistic, haptic rendering requires that the deformed object *feels* realistic, as well. While the update rate in graphics rendering needs to be around 20–30 Hz to look realistic, the update rate in haptic rendering needs to reach 1 kHz to feel realistic. Therefore, haptic rendering has a much more stringent requirement than graphics rendering for high rendering quality, which is essential to many applications that simulate manipulations or interactions in the real physical world.

In order to achieve a high rendering rate, existing approaches on haptic rendering often apply certain simplifications to the physically based deformable models used in graphics rendering, such as mass-spring-damper models and continuum models, and focus on simple contact cases.

In methods based on mass-spring-damper models [4]–[10], an elastic object is constructed by applying a mass at each point of a mesh and using springs to link the points as edges and diagonals. Elastic forces and damping forces act on mass points as internal forces, and gravity and other possible forces act on them as external forces. A linear strain model is often used as an approximation of the real nonlinear model of the deformable object. Such mass-spring-damper models have been widely used in many applications. As simple physical models with well-understood dynamics, they are easy to construct and can be used for interactive and even real-time simulation. However, the mass-spring-damper models have drawbacks. The physical accuracy of modeling is often not sufficient. For example, incompressible volumetric objects or thin surfaces that are resistant to bending are difficult to model as mass-spring systems. The models are linear, and in order to simulate nonlinear force responses, it is necessary to use a precise integration mechanism, such as the finite element method (FEM), but such a method generally cannot provide update rates that are sufficient for haptic interactions.

As for continuum models, which include models based on the FEM [11]–[21], the finite difference method [22], the boundary element method [23], and the long element method [24]–[26], perceptually acceptable performance can be achieved only by further simplifications or adaptive methods. With the aid of precalculation and multiresolution approaches [27], deformation of more complicated objects can be simulated in real-time.

However, to achieve real-time results, most of the current approaches to contact force/torque modeling involving deformable objects are focused on single-point contacts [21], [28]–[31], single-area contacts [23], [32], [33], and localized deformation [13]. Little is done to model deformable objects in complex contact states involving multiple contacts, in compliant motions with friction and undergoing global deformation.

We introduce a novel approach to contact modeling and deformation modeling between a rigid object and a deformable object in interactive environments. Our approach can handle complex contact states with multiple contact regions and compliant motions with friction. We use the general Duffing equation [34] as a foundation to simulate nonlinear contact forces from a deformed object due to contact. The Duffing equation is one of the standard models for studying nonlinear systems subject to external forces. It is well studied, relatively simple, and yet adequately powerful to model very complex behaviors [35]. This model is particularly suitable for modeling the nonlinear stiffness of biomaterials, as common in surgical simulations. Contact forces of different types of deformable objects (i.e., elastic, plastic, etc.) can be simulated by changing the related parameters, which can be obtained through precalculations [13].

We model both global deformation of the entire elastic object and local deformation within neighborhoods of contact regions. In order to handle the global shape change due to deformation, we introduce a novel beam-skeleton model to compute the distribution of stresses and strains of a deformed elastic object at certain anchor points, defined on the original undeformed surface of the object. Based on this model, we further introduce fast computation of global shape change through an interpolation method that achieves minimization of elastic energy. We next take into account the nonlinear effects within local neighborhoods of contact regions and the effects of different areas of contact regions (under the same force) on shape change.

This paper substantially extends our prior work [36] with a modified beam-skeleton model more apt for modeling force and deformation under multiple contact regions and applicable to deformable objects of nonhomogeneous (i.e., piecewise homogeneous) material. It delivers a more detailed treatment of contact detection, adds a description of contact force calculation of a nonpoint contact region, and presents a more thorough method for friction handling and compliant motion.

The overview of our approach for contact and shape deformation modeling and rendering in an interactive environment is shown in a flowchart in Fig. 1.

The rest of the paper is organized as follows. In Section II, we introduce basic assumptions of our approach. In Sections III and IV, we describe our methods for contact detection and contact force modeling. In Section V, we present our basic beam-skeleton model for shape deformation modeling. In Section VI, we describe our strategy for dealing with multiple contact regions. We present some implementation results in Section VII, and conclude the paper in Section VIII.

II. Basic Assumptions

This section presents assumptions and associated terminology.

A. Objects and Material

We focus on modeling the interactions between a rigid object held by a human user and a deformable object. Depending on material properties, deformable objects can be categorized into many types [37]. In this paper, we focus on deformable objects that are made of isotropic elastic material.

We use a mesh model representation for the geometry of the rigid held object. We also use mesh models to represent the elastic object in its original undeformed shape and in deformed shapes. The elastic object we consider here is convex when undeformed, and there is a parametric model of the originally undeformed surface.

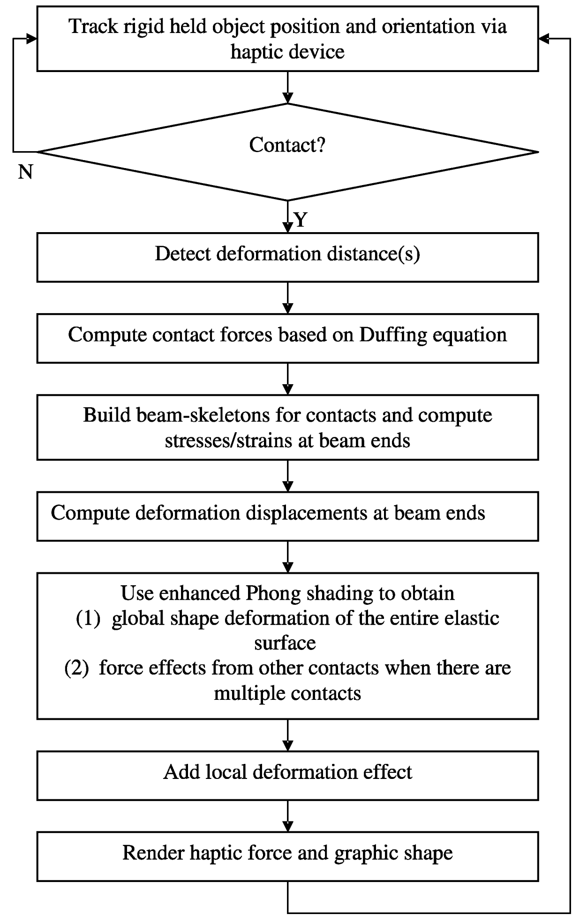


Fig. 1. Overview of approach (one time step per loop).

We assume that the force exerted to the held object from the human user is applied to the mass center of the held object. This assumption is useful later for estimating the distribution of contact pressure (see Section IV-C and Appendix III).

B. Contacts and Compliant Motion

We define a single *contact region* as a cluster of contact points S such that the distance between a contact point in S and its nearest neighboring contact point in S is less than a threshold r_{th}. A contact point outside S is considered belonging to another contact region, and there can be multiple contact regions in general.

We only focus on cases where each single contact region is relatively small so that within the contact region, the first partial derivatives of the originally undeformed surface of the elastic object hardly change. A contact region may consist of just a single contact point.

For any point on the elastic object outside a certain immediate neighborhood R of a contact region, we consider its deformation as caused by the stresses and strains spread to it from the contact region as a function of the contact force, and call such a deformation *global deformation*. The stress causing the global deformation can be considered as linearly distributed.

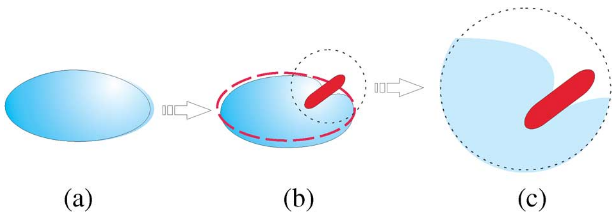


Fig. 2. Example of shape deformation. (a) Originally undeformed elastic object. (b) Global deformation over the whole elastic object (the red dash line shows the surface before deformation). (c) Local neighborhood deformation.

For points inside the neighborhood R of the contact region, we take into account the nonlinear effect of deformation, and also that the shape deformation is not only caused by the contact force but also by the size of the contact region; the greater the size, the smaller the pressures are under the same force, and thus the smaller the deformation. We therefore modify the shape deformation inside R accordingly, and call the result *local neighborhood deformation* (Fig. 2).

There can exist multiple contact regions between the rigid object and the elastic object at the same time, but multiple contact regions are assumed to be formed one by one. The rigid object can move compliantly on the surface of the elastic object, but when there are multiple contact regions, no compliant motion is considered between the rigid object and the elastic object.

C. Stable Equilibrium Configurations

We only consider modeling contact forces caused by quasi-static collisions and compliant motion. This means that motions considered are slow enough such that only deformations occurring at stable equilibrium configurations need to be considered, where the elastic energy is minimized. This provides an effective discretization of the otherwise continuous force and shape changes on the elastic object in contact.

III. Real-Time Collision Detection

Although collision detection between rigid objects is well investigated, collision detection involving dynamically deforming objects in an interactive environment is not a fully solved problem. Teschner *et al.* [38] surveyed recent approaches to collision detection involving deformable objects. These approaches are mainly based on bounding volume hierarchies (BVH) [39], stochastic methods [32], [40], distance fields [41], [42], spatial partitioning [43], and image-space techniques [44]. However, haptic interaction with deformable objects require both real-time (in kHz) and exact contact detection with distance information. Currently, there is no fast contact detection algorithm that can obtain accurate penetration distances when there are multiple contacts between a globally deformable object and another object.

Our strategy is to treat contact detection between a rigid body and an elastic object as if between two rigid bodies at the time of contact, and then dynamically change the surface model of the elastic object to reflect the shape changes due to contacts. In each time step i, contact detection is characterized by one of the following three phases.

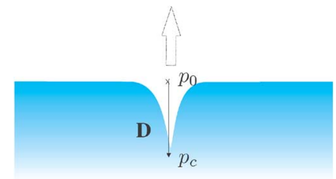


Fig. 3. Contact force simulation. A single-point contact with normal compression (the hollow arrow indicates contact force direction).

In the first phase, no contact has yet happened between the two objects, and thus the elastic object does not change its shape. Here collision detection can be considered as just between two rigid objects (in mesh models). We use a real-time collision detection package SOLID [45] to detect the contact and provide the distance information between the two objects.

The second phase is marked by the transition from no contact to contact between the two objects, i.e., at least one contact point is established in the current time step i. Once that happens, one contact region is formed and recorded.

The third phase describes the situation when the two objects were already in contact in the previous time step $i-1$ and remain in contact in the current time step i, but the contact region(s) may change. There are two possible changes: an existing contact region changes because the contact points in it change; and a new contact region is formed. How to detect new contact regions and track multiple contact regions are detailed in Appendix I.

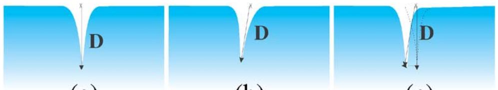
After all contact regions at time step i are determined, contact forces and the corresponding shape change of the elastic object can be computed and rendered haptically and graphically for time step i in real-time, as detailed in Sections IV–VI. The shape of the deformable object is then updated, and the updated shape is used as the input shape for contact detection at the next time step $i + 1$. In other words, in each time step, the shape update of the deformable object is relative to the shape obtained in the previous time step.

IV. Contact Force Modeling for Haptic Rendering

In this section, we first apply the general Duffing equation to provide a basic nonlinear contact force model for a single-point contact caused by pressing the rigid object normally to a face of the elastic object (Fig. 3). Then we extend the method to model other cases of single-point contact or single-region contact.

A. Relation Between Contact Force and Deformation Displacement

One of the commonly used nonlinear equations for characterizing the behavior of nonlinear mechanical, electrical, and chemical systems is the Duffing equation. We can use the Duffing equation to characterize the nonlinear force response of an elastic object in the basic single-point contact case at a quasi-static state, as shown in Fig. 3, where p_0 indicates the position of the contact point before the contact and the resulted deformation, which we call the *origin of deformation*, and $\mathbf{D}$ is the distance from p_0 to the point of maximum deformation p_c, which we call the *deformation displacement vector* with magnitude D. From the Duffing equation, we can derive the



(a) (b) (c)

Fig. 4. Contact force simulation. (a) Contact is normal to the elastic surface. (b) Skewed deformation. (c) Compliant motion with deformation.

following relation between the contact force response $\mathbf{F}_c$ and $\mathbf{D}$:

$$||\mathbf{F}_c|| = m\left|\omega_0^2 D + \frac{3\beta_0^2\epsilon}{4}D^3\right| \tag{1}$$

where m is the mass of the rigid object. ω_0, $\beta_0^2\epsilon$ are the parameters of linear and nonlinear restoring terms in the Duffing equation. All of them are constants and can be acquired by measurement beforehand. A detailed derivation can be found in Appendix II.

B. Skewed Deformation and Compliant Motion

Now we extend our basic method of contact force simulation to point contact cases where the direction of deformation is not normal to the surface of the elastic object before this deformation so that the held rigid object may get stuck or perform compliant motion on the elastic object.

When the direction of deformation is not normal to the surface of the elastic object before this deformation Fig. 4(b), the deformation displacement vector $\mathbf{D}$ can be decomposed into tangential and normal components $\mathbf{D}_t$ and $\mathbf{D}_n$, respectively. The force response due to deformation along the normal direction $\mathbf{F}_{cn}$ can be computed from (1) with $D = ||\mathbf{D}_n||$, pointing to the direction against $\mathbf{D}_n$.

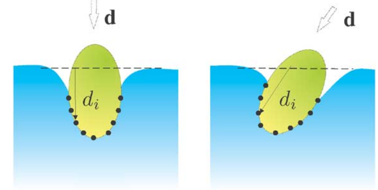
Now we need to detect whether the rigid body is stuck or performs a compliant motion tangentially along the contact surface of the elastic object. First, assume that the rigid body is stuck at the current time step i due to the tangential deformation force response $\mathbf{F}_{ct}$ from the elastic object, which can be computed from (1) with $D = ||\mathbf{D}_t||$, pointing to the direction against $\mathbf{D}_t$.

According to [46], the maximum friction from objects of different elasticity is proportional to $||\mathbf{F}_{cn}||^\beta$, $2/3 \le \beta \le 1$, in the empirical equation $f_{\max} = K||\mathbf{F}_{\rm cn}||^\beta$ where the coefficient K and β were given in [47] for various deformable materials. For a truly elastic solid, $\beta = 2/3$.

Now our assumption that the rigid object is stuck is correct if one of the following two conditions holds:

- C1: $||\mathbf{F}_{ct}|| \le f_{\max}$, and in the previous time step $i-1$, the rigid body was either stuck at contact or not in contact with the elastic object;
- C2: $||\mathbf{F}_{ct}|| \le f_{\max}$, the rigid object was in compliant motion at time step $i-1$, but $\mu_D||\mathbf{F}_{cn}|| \ge f_{\max}$, where μ_D is the dynamic friction coefficient.

The C1 condition indicates that the rigid object remains stuck in the current time step i. The C2 condition indicates that the rigid object stops and gets stuck in the current time step i from the motion in time step $i-1$. This is because it is impossible for the dynamic friction to be greater than $f_{\max}$ if the rigid object is still in motion. As the rigid object is stuck, only a skewed deformation happens (see Fig. 4(b). Note that the shape of deformation is modeled in Section V). The total contact force response from the elastic object to the rigid body is $\mathbf{F}_c = \mathbf{F}_{cn} + \mathbf{F}_{ct}$.



(a) (b)

Fig. 5. Examples of single-region contact and contact region discretization ($\mathbf{d}$ is the direction of deformation). (a) Normal deformation. (b) Skewed deformation.

Otherwise, the assumption is incorrect, that is, the rigid object in fact moves in time step i if one of the following two conditions holds:

- C3: $||\mathbf{F}_{ct}|| \le f_{\max}$, the rigid body was in compliant motion in time step $i-1$, and $\mu_D||\mathbf{F}_{cn}|| < f_{\max}$;
- C4: $||\mathbf{F}_{ct}|| > f_{\max}$.

In both cases, $\mathbf{F}_{ct}$ needs to be recalculated to match the correct scenario that the rigid object is in motion: if the contact point p_c (on the held rigid object) has moved tangentially from time step $i-1$ to time step i with a distance Δd, then to model the effect of compliant motion, we also shift p_0 (i.e., the origin of the deformation distribution) the distance Δd to obtain its new position [as shown in Fig. 4(c)]. Then we recalculate the dynamic friction $\mathbf{F}_{ct}$ as $||\mathbf{F}_{ct}|| = \mu_D||\mathbf{F}_{cn}||$ if C3 holds. If C4 holds, then $||\mathbf{F}_{ct}|| = \min(\mu_D||\mathbf{F}_{cn}||, f_{\max})$. Note that if $\mu_D||\mathbf{F}_{cn}|| > f_{\max}$, this shows again an impossible case. It means that the normal displacement of the contact point p_c is impossibly large. Therefore, $||\mathbf{F}_{cn}||$ needs to be recalculated as $||\mathbf{F}_{cn}|| = f_{\max}/\mu_D$, and based on which, the normal displacement of p_c needs to be adjusted to satisfy (1).

The total contact force response from the elastic object to the rigid body corresponding to compliant motion of the rigid body can be obtained as the sum of the recalculated $\mathbf{F}_{ct}$ and (sometimes recalculated) $\mathbf{F}_{cn}$.

C. Single Region Contact

A single-region contact is formed by a continuous set of contact points. The total effect of contact forces can be obtained by integrating contact forces generated on contact points (or infinitesimal contact areas) over the whole contact region. We discretize the force integration as the summation of contact forces at a number of evenly distributed contact points with different displacements of deformation, as shown in Fig. 5. To achieve real-time processing, the discretization can be simply based on the vertex points of the mesh model of the contact region of the rigid object, provided that these vertex points are evenly distributed on the mesh. The contact force response F_i at each contact point p_i can be calculated from the general Duffing equation based on its deformation displacement d_i (Fig. 5) and the mass m_i distributed on it (see Appendix III). Summing up all F_i gives the total force F against the direction of deformation $\mathbf{d}$.

In the case of a normal deformation, Fig. 5(a), the computed F is *the* total contact force response F_c along the normal of the surface of the elastic object before this deformation.

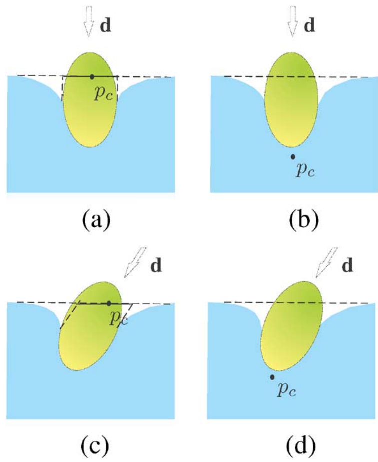


Fig. 6. Examples of equivalent contact point p_c. (a) Before normal deformation. (b) After normal deformation. (c) Before skewed deformation. (d) After deformation.

However, in the case of skewed deformation [see Fig. 5(b)], the computed $\mathbf{F}$ with magnitude F along direction $-\mathbf{d}$ should be further decomposed to a normal component $\mathbf{F}_{cn}$ (i.e., normal to the surface of the elastic object before this deformation) and a tangential component $\mathbf{F}_{ct}$. Depending on the magnitude of $\mathbf{F}_{ct}$, following the same analysis as presented in Section IV-B, we can determine whether the held rigid object gets stuck or performs compliant motion as the single-region contact occurs, and obtain the corresponding total contact force $\mathbf{F}_c$.

Next, we can find an *equivalent point contact* to the single-region contact in that the contact force response to that point contact is the same as the total contact force response of the single-region contact $\mathbf{F}_c$. With known F_c and m (m is the mass of the rigid object, which can be considered as concentrated on the equivalent contact point), the equivalent deformation displacement D of the equivalent contact point can be obtained from (1). The position of the equivalent contact point p_c before deformation can be considered as at the geometric center of the projection of the contact region on the surface of the elastic object before this deformation along the deformation direction $\mathbf{d}$, called again the *origin of deformation*. See Fig. 6. Using such an equivalence of a point contact to the original single-region contact simplifies the graphical shape rendering of the deformed elastic object (see Section V).

V. Shape Deformation Modeling

In Section IV, contact force response from a contact point or a contact region of the elastic object to the rigid object is modeled. In the case of a single-region contact, based on the Saint-Venant principle [48], the contact forces integrated over the whole contact region will have the same effect as a total contact force at a single equivalent point contact. Thus, we use the equivalent point contact (obtained in Section IV-C) to compute the deformation just as in the case of a single-point contact.

Now we consider how to model the shape deformation occurring on the elastic object due to contact. In the following, we first address the modeling of global deformation outside local neighborhoods of contact regions (Sections V-A–V-E) and next consider effects of local deformation inside each local neighborhood of a contact region (Section V-F).

A. Anchor Points

Because the elastic object without deformation consists of smooth (flat or curved) surface patches, which may be bounded by smooth (straight-line or curved) edges and vertices, its deformed surface patches and edges along with the corresponding stresses and strains should also be smooth, except at bounding vertices and the contact (or equivalent contact) point to minimize elastic energy. Additionally, the stresses and strains reach extremal values at other extremal points of the elastic surface.

We use the term anchor points to mean those vertices or extremal points: an *anchor point* is either a natural vertex, a curvature discontinuous point, a discontinuous point of the first derivative of curvature, a point with local minimum or maximum curvature, or an inflection point with zero curvature [49]. For a surface of revolution, which is a surface generated by rotating a 2-D curve about an axis, no such point exists. Instead, it has *anchor circles* formed by rotating the anchor points of the initial 2-D curve. Since a complete circle has azimuthal symmetry, we can draw any two perpendicular axes through the center of the circle to obtain four intersection points (of the axes and the circle). We choose such four points on an anchor circle as anchor points. This is a reasonable choice based on the Four Vertex Theorem in differential geometry [50]: A closed embedded smooth plane curve has at least four vertices, where a vertex is defined as an extremum of curvature. As for a sphere, which is a special case of revolution surface, we use six points that are intersections of the sphere with any three perpendicular axes set at its center as its anchor points.

B. Beam-Skeleton Model

Since the global shape deformation of the elastic object is nearly linear, in order to compute it efficiently in real time, we introduce a novel modeling approach based on the Bernoulli–Euler bending beam theory as follows:

1) once a contact is formed, establish a beam-skeleton (to be specified below) on the elastic object with beams ending at anchor points, called beam ends;
2) compute values of stresses and strains at each beam end of the beam-skeleton;
3) compute values of strains over the entire surface via smooth interpolation of the values at beam ends to obtain the shape of the elastic object after global deformation.

A beam-skeleton with respect to a contact is established as follows: once a contact is formed, between the origin of deformation p and each anchor point p_i $(i = 1, 2, \ldots)$ on the elastic object, a beam i is established with the central line being the line connecting p and p_i. The beam parameters are determined by the physical properties as well as the smooth surface properties of the elastic object. The length of beam i is l_i, which is the distance between p and p_i if p_i is outside the local neighborhood R of the contact. If p_i is inside the local neighborhood R of the contact, then $l_i = r_R$, where $r_R > 0$ is the radius of R centered at p. The cross-section area of beam i is s_i. The computation of s_i is described in Appendix IV. The collection of

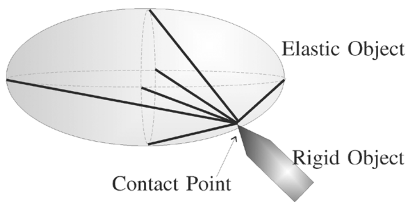


Fig. 7. Beam skeleton (in solid lines) on an elastic ellipsoid object.

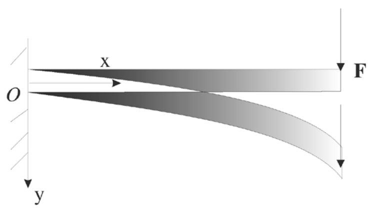


Fig. 8. Schematic of beam bending.

such beams forms a beam skeleton. Fig. 7 shows an example of a beam skeleton on an elastic ellipsoid object.

Note that when the contact moves in compliant motion, the beam-skeleton moves accordingly, and the beam length l_i for each beam will change. When the contact is broken, the beam-skeleton disappears.

The above beam-skeleton model is quite powerful with the underlying beam-bending theory, which can compute general deformation effects resulted from extension/compression, bending, and twisting beams, as detailed in the following subsections.

C. Deformation Computation for Beams of Homogeneous Objects

First we describe how deformation is computed for a beam of length l with cross-section area s bent at one end, with the other end fixed based on the Bernoulli–Euler bending beam theory [51]. For a homogeneous object, such a beam is made of homogeneous material.

Establish the beam coordinate system as $O - xyz$ as shown in Fig. 8, where the origin is set at the center point of the fixed end of the beam, and the x axis is along the central line of the beam before it is bent and pointing to the other end of the beam. The y axis follows the bending force direction, and the z axis is orthogonal to both x and y axes following the right-hand rule. A point p on the central line of the beam before it is bent has coordinates (x, 0, 0). Once the beam is pressed at the end that is not fixed, the beam bends, and the new coordinates of p is (x', y', z') satisfying

$$x' = x - \delta x \qquad y' = \frac{F_y}{EI_z}\left(\frac{1}{2}lx^2 - \frac{1}{6}x^3\right) \qquad z' = 0$$

where $\delta x = x - (x^2/\sqrt{x^2 + y'^2}$ is the modification item due to possible large deformation. When y' is small, $\delta x \approx 0$. E is the Young's modulus, and I_z is moment of inertia with respect to the z axis. At the end of the beam where $x = l$, the relation between the external force F_y normally applied to the beam end and the deformation distance y' is

$$F_y = \frac{3EI_z}{l^3}y'. \tag{2}$$

We can relate F_y to the stresses on the beam

$$\begin{aligned}
\sigma_x &= -\frac{F_y}{I_z}(l - x)y \\
\sigma_y &= \sigma_z = \tau_{yz} = 0 \\
\tau_{yx} &= \frac{1}{2(1+\nu)}\frac{F_y}{I_z}\left[\frac{\partial \phi_1}{\partial y} + \nu z^2 - (1+\nu)y^2\right] \\
\tau_{zx} &= \frac{1}{2(1+\nu)}\frac{F_y}{I_z}\frac{\partial \phi_1}{\partial z}
\end{aligned}$$

where σ's are stresses, τ's are shear stresses, I_z is the moment of inertia, and ϕ_1 is the bending function of the deformable object depending on the shape of the cross section of the beam; for different shapes, ϕ_1 is different.

We call the above case where the external force is normal to the beam a *simple bending case*.

In general, the external force applied to a beam at one end is not necessarily normal to the beam central line. In such a case, based on the Saint-Venant principle, we can decompose the force into a normal force and a tangential one and decompose the problem of relating the external force to beam deformation into two simpler cases: one is the above simple bending case, and the other is a simple compression/expansion case, which we describe in the following.

For the simple compression/expansion of the beam caused by a tangential external force F_x, we can relate F_x to the tangential deformation (i.e., compression or expansion) Δx at the beam end $x = l$ with the following equation:

$$F_x = \frac{Es}{l}\Delta x. \tag{3}$$

We can also relate F_x to the stresses of the beam

$$\sigma_x = F_x/s \qquad \sigma_y = \sigma_z = \tau_{xy} = \tau_{xz} = \tau_{yz} = 0.$$

The total stresses at a point of a beam are the vector sums of the stresses from the simple bending case and simple compression/expansion case. Note that the tangential deformation Δx for the calculation of total stresses is the changed beam length due to the deformation minus the δx due to bending.

In our application, the force $\mathbf{F}$ applied to a contact or equivalent contact point from the rigid object to the elastic object can be viewed as applied to the common end of a beam skeleton consisting of n beams. The force $\mathbf{F}$ can be obtained as opposing the contact force response from the elastic object (as computed in Section IV) with the same magnitude. Establish a coordinate frame $O_i - x_i y_i z_i$ for each beam i $(i = 1, \ldots, n)$ of the beam skeleton such that the origin O_i is located at the anchor point of beam i, which is at the other end of the beam i, x_i axis is along the beam central line before it is deformed and pointing to the common end, y_i is on the plane determined by the x_i axis and the force $\mathbf{F}$, and the z_i axis is orthogonal to x_i and y_i axes following the right-hand rule.

Now we can view $\mathbf{F}$ applied to the same (contact) point of the common end of all the beams in the skeleton as the sum of the forces $\mathbf{F}_i$ applied to each beam i at the same point, so that the deformation occurred at each beam can then be computed separately. Notice that $\mathbf{F}$ does not have a z_i component in the beam i's frame. Let F_{x_i} and F_{y_i} be the x_i and y_i components of $\mathbf{F}_i$. From (2) and (3), we can obtain the following relations among all $\mathbf{F}_i$'s, and the relation between each $\mathbf{F}_i$ and the deformation of beam i at the common end of the beam skeleton expressed in beam i's frame:

$$\begin{aligned} F_{x_1} &: \ldots : F_{x_i} : \ldots : F_{x_n} \\ &= \frac{\Delta x_1 s_1}{l_1} : \ldots : \frac{\Delta x_i s_i}{l_i} : \ldots : \frac{\Delta x_n s_n}{l_n} \end{aligned} \tag{4}$$

$$\begin{aligned} F_{y_1} &: \ldots : F_{y_i} : \ldots : F_{y_n} \\ &= \frac{y_1' I_{z,1}}{l_1^3} : \ldots : \frac{y_i' I_{z,i}}{l_i^3} : \ldots : \frac{y_n' I_{z,n}}{l_n^3} \end{aligned} \tag{5}$$

where s_i, l_i, and $I_{z,i}$ are the cross-section area, the length, and the moment of inertia of beam i, respectively.

Additionally, the sum of all $\mathbf{F}_i$'s, after transforming with respect to the world coordinate system w, should equal $\mathbf{F}$ expressed in w

$$\sum_{i=1}^{n} {}^{w}\mathbf{R}_i \mathbf{F}_i = \mathbf{F} \tag{6}$$

where ${}^{w}\mathbf{R}_i$ is the rotational transformation matrix from frame i to frame w. From the $2n$ (4), (5), and (6), $\mathbf{F}_i$ can be solved for each beam i.

With the above method, we can compute the stresses at the fixed end of each beam centered at an anchor point of the elastic object. Now imagine the fixed end of each beam is no longer fixed, in which case, the effect of the stresses will make the corresponding anchor point move to a new position. The position change can be considered as the deformation at such an anchor point, which can be computed from those stresses.

Now we describe how to compute strains from stresses and subsequently compute the deformation at each beam end in the beam-skeleton.

The relation between the strain ϵ at point $\{x, y, z\}$ and the deformation displacement $\{u, v, w\}$ at this point can be represented as

$$\epsilon = \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \end{Bmatrix} \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} \tag{7}$$

and the relation between the strain and stress at point $\{x, y, z\}$ can be represented as

$$\epsilon = \mathcal{D}^{-1}\sigma \tag{8}$$

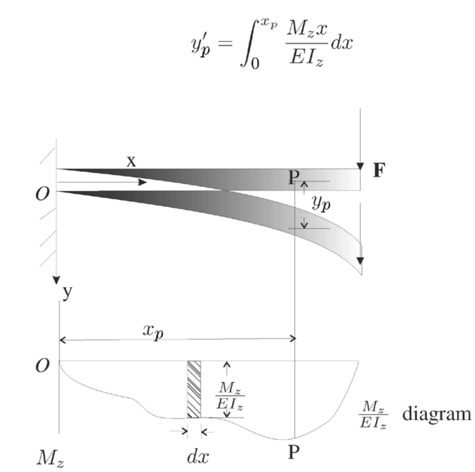


Fig. 9. Derivation and interpretation of the moment-area method for determining deflections of a nonhomogeneous beam.

where $\mathcal{D}^{-1}$ is the inverse matrix of the elastic coefficient matrix $\mathcal{D}$ and

$$\mathcal{D}^{-1} = \frac{1}{E} \begin{bmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{bmatrix} \tag{9}$$

where E is Young's module, and ν is Poisson's ratio.

Given an anchor point with coordinates $\{x, y, z\}$ and its stresses, we can get its strains from (8). Next, its displacement due to deformation $\{u, v, w\}$ can be solved from these strains using (7).

D. Deformation Computation for Beams of Nonhomogeneous Objects

Now we extend the above deformation computation method to objects of nonhomogeneous materials. For such an object under contact, a beam skeleton can be established in the same way as in the case of a homogeneous object, but the computation of deformation for a beam of nonhomogeneous material is different from that in the homogeneous case. Consider a beam of nonhomogeneous material with one end fixed, and establish a beam coordinate system in the same way as in the case of a homogeneous beam. Let p be a point on a beam and (x_p, 0, 0) be the position of p before the beam is bent. Based on the *second moment-area theorem* and its corollary on deflection of any point on a beam with one end fixed [52], when the beam is bent at the other end, the magnitude of tangential deviation y_p' at point p of the beam can be obtained from the moment of the area of the $M_z/(EI_z)$ diagram (where M_z is the bending moment) between the fixed end and p, as shown in Fig. 9, and satisfies

$$y_p' = \int_0^{x_p} \frac{M_z x}{EI_z} dx. \tag{10}$$

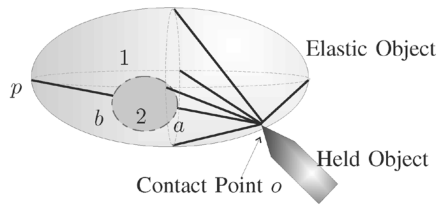


Fig. 10. Beam skeleton of a contact on an nonhomogeneous elastic ellipsoid.

Since each nonhomogeneous beam can be viewed as consisting of different homogeneous segments, as long as the number of such segments is relatively small, the above integral can be computed very quickly.

As an example, consider the case shown in Fig. 10, where a ball of different material exists inside an ellipsoidal object so that the object is nonhomogeneous with two homogeneous regions, named region 1 and region 2. The two regions have Young's modulus E_1 and E_2 and moments of inertia I_{z1} and I_{z2}, respectively. A nonhomogeneous beam op crosses region 2 at a and b. For simple bending of this beam at the end point p, the displacement y'_p can be obtained from (10) in terms of the bending force magnitude F_y. Conversely, F_y can be written as a function of y'_p

$$F_y = \frac{y'_p}{\frac{1}{E_1 I_{z1}}\left(\frac{l^3}{3} - k\right) + \frac{1}{E_2 I_{z2}} k} \tag{11}$$

where l is the length of the beam, $k = (\alpha l)/2 - \beta$ with $\alpha = l_b^2 - l_a^2$, $\beta = (l_b^3 - l_a^3)/6$, and l_a and l_b are the lengths of oa and ob, respectively. l is the length of the undeformed beam op. Stresses can also be obtained in segments.

For simple compression/expansion of the beam op, we have

$$F_x = \frac{s}{l}\left(E_1(\Delta x_{oa} + \Delta x_{bp}) + E_2 \Delta x_{ab}\right) \tag{12}$$

where Δx_{oa}, Δx_{ab}, Δx_{bp} are the tangential deformations of each beam segments. Stresses can be calculated in the same way as in the homogenous case.

If the nonhomogeneous beam op is the ith beam while the other beams are homogeneous inside region 1, based on (12) and (11), the force distribution (4) and (5) can be modified by replacing the ith right-hand terms, respectively, with the following:

$$\begin{aligned} F_{x_1} &: \ldots : F_{x_i} : \ldots : F_{x_n} \\ &= \frac{\Delta x_1 s_1}{l_1} : \ldots : \frac{\Delta x_{i-1} s_{i-1}}{l_{i-1}} : \\ &\quad \frac{\left((\Delta x_{oa} + \Delta x_{bp}) + \frac{E_2 \Delta x_{ab}}{E_1}\right) s_i}{l_i} : \\ &\quad \frac{\Delta x_{i+1} s_{i+1}}{l_{i+1}} : \ldots : \frac{\Delta x_n s_n}{l_n} \end{aligned} \tag{13}$$

$$\begin{aligned} F_{y_1} &: \ldots : F_{y_i} : \ldots : F_{y_n} \\ &= \frac{y'_1 I_{z,1}}{l_1^3} : \ldots : \frac{y'_{i-1} I_{z,i-1}}{l_{i-1}^3} : \frac{y'_i I_{z,i}}{(l_i^3 - 3k) + \frac{3kE_1}{E_2}} : \\ &\quad \frac{y'_{i+1} I_{z,i+1}}{l_{i+1}^3} : \ldots : \frac{y'_n I_{z,n}}{l_n^3}. \end{aligned} \tag{14}$$

If more beams are nonhomogeneous, i.e., crossing both two regions, their corresponding terms in the right-hand side of (13) and (14) should be replaced similarly as in the case of the ith terms. With the bending force solved for each beam based on the modified force distribution equations, stresses can be computed.

Note that if the external surface of the elastic object consists of different homogeneous patches corresponding to different homogeneous pieces of the object, beam ends may be from different homogeneous patches. In this case, the extended beam-skeleton model above captures the different stress effects on anchor points from different homogeneous patches.

Subsequently, strains and deformation displacement at each anchor point can be computed in the same way as in the case of homogeneous objects, and the anchor points are still points where the deformation has extreme values.

E. Global Deformation Rendering

With the displacements of all beam ends of the elastic object and the displacement of the contact (or equivalent contact) point due to deformation, we obtain the globally deformed shape of the entire elastic object by an interpolation method extending the Phong shading method [53]. Phong shading is used for linear interpolation of vectors at vertices bounding a polygon across internal points of the polygon.

In our case, however, the elastic object can have a general surface with curved features with or without deformation. Let P be the set of beam ends on the surface of the elastic object before the current deformation. Such a surface can be partitioned by curves connecting each pair of points in P into surface patches. Let P_k be a subset of P corresponding to a single-surface patch k. Given the deformation displacement vectors of points in P_k, we extend Phong shading to obtain a linear interpolation of the deformation displacement vectors across the surface patch k as the following.

> Let θ_i denote the angle between the direction of displacement and the outward normal direction of point i in P_k of surface patch k, called the *displacement angle* of point i; the directions of displacement vectors across the surface patch k are obtained by linearly interpolating the displacement angles of the points in P_k, and the magnitudes of displacement vectors across the surface patch k are obtained by linearly interpolating the magnitudes of displacement vectors of points in P_k.

We obtain the deformed shape of the entire elastic object by performing the above shading on all surface patches. Note that since we do graphical rendering of an object based on its polygonal mesh approximation, the interpolated points of deformation shading of a face can be simply the corresponding mesh points of the face.

The global shape deformation of the elastic object obtained from interpolating these displacements is smooth and satisfies that the closer a surface point to the contact point or the equivalent contact point, the greater the deformation change is at this point.

Once the deformed shape of the elastic object is obtained in terms of new positions of its mesh points, the triangle normals

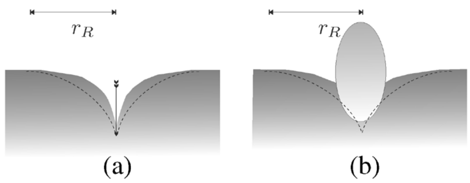


Fig. 11. Two examples of local neighborhood shape deformation (where the dotted lines indicate the results from global deformation). (a) Point contact. (b) Region contact.

and normals at mesh points are then obtained in standard ways of graphics rendering.

F. Local Neighborhood Deformation

The concentrated effect of load introduces very high stresses in localized regions of a contact [54]. The shape deformation is nonlinear inside the local neighborhood R of a contact region, and therefore, we need to modify the result of global shape deformation by adding the nonlinear effect inside R.

For the case of a single-point contact with the origin of deformation p_0, we define a *modification factor* $w_p = 1+C_n \log((r+C_r)/(r_R+C_r))$ for each noncontact point p inside R to capture the nonlinear effect, where r is the distance between p before its deformation caused by the contact and p_0, r_R is the radius of the neighborhood R centered at p_0, and C_n and C_r are constants that represent the nonlinear coefficients. Now, we multiply w_p to the displacement of global deformation at p to obtain its modified displacement. The result is the combined effect of global shape deformation with local nonlinear modification.

For the case of a single-region contact, the effect of the contact area on shape deformation in the neighborhood R also needs to be taken into account. Since the number of mesh points n in the contact region (see Appendix III) is proportional to the area of the contact region, we further modify our modification factor at each noncontact point p as $w'_p = (1+C_n \log((r+C_r)/(r_R+C_r)))/(1+\log n)$ to also capture the effect of the contact area on deformation: the larger the area, the shallower the deformation. By multiplying w'_p to the displacement of global deformation at p, we obtain the deformation displacement at p. Fig. 11 shows two examples of local neighborhood shape adjustment for a single-point contact and a single-region contact, respectively. As shown, the deformation is shallower for the single-region contact than the single-point contact when the equivalent point contact is at the same depth as the single-point contact.

VI. Dealing With Multiple Contact Regions

At time step k, suppose there are m contact regions as the result of our contact detection algorithm (Section III). We view them as a sequence, S_i, $i = 1, \ldots, m$, ordered by the times of their creation.[1] The effect of contact regions formed before S_i, i.e., $S_1, \ldots, S_{i-1}$, is taken into account by the fact that S_i is formed with respect to the already deformed shape of the elastic object under $S_1, \ldots, S_{i-1}$, call it Shapee$_{i-1}$. Note that Shape$_0$ is the originally undeformed shape of the elastic object.

[1]From Section II-B, it is assumed that only one contact region is created at each time step.

We establish a beam skeleton B_i with respect to each contact region S_i as follows: each beam in B_i is from the origin of deformation p_i of S_i to an anchor point of the elastic object under Shape$_{i-1}$.

Now we describe how to compute the deformation force effect at S_i and the deformed shape of the elastic object under those m contacts below.

A. Contact Force Modeling

The computation of contact force effect F_{S_i} to the rigid object at contact region S_i depends on the existence of other contact regions. Treating $Shape_{i-1}$ as the "undeformed" shape for S_i, then we can compute a force $\mathbf{F}_{ii}$ in the same way as the force $\mathbf{F}$ is computed in Section IV-C. Note that we do not assume compliant motion here. $\mathbf{F}_{ii}$ already takes into account the effects of contact regions $S_1, \ldots, S_{i-1}$ formed before S_i as they are captured by Shape$_{i-1}$.

Let $\mathbf{F}_{ji}$, where $j = i+1, \ldots, m$, be the force effect on S_i from the contact region S_j formed after S_i. $\mathbf{F}_{ji}$ can be computed from B_j by integrating the stresses at contact points of S_i due to S_j. The stresses at those contact points can be obtained by interpolation based on stresses computed from B_j at the related anchor points (i.e., the anchor points bounding the face where S_i lies) following the same method as that for interpolating displacements in Section V-E.

The total contact force effect at S_i is thus $\mathbf{F}_{S_i} = \mathbf{F}_{ii} + \Sigma_j \mathbf{F}_{ji}$ (of course, all forces are transformed to the same coordinate frame before addition).

The total contact force effect under m contact regions is $\mathbf{F}_c = \sum_{i=1}^{m} \mathbf{F}_{S_i}$.

B. Shape Deformation Modeling

Now we describe how to obtain Shape$_i$, $i = 1, \ldots, m$, based on Shape$_{i-1}$ at time step k. The deformation displacement caused by S_i at each anchor point can be computed from B_i in the same way as described in Sections V-C or V-D. Note that the deformation displacement at the (equivalent) contact point of each contact region S_k formed prior to S_i (i.e., $k = 1, \ldots, i-1$) is considered zero, because these contact points are fixed by the rigid object. Now, the global deformation caused by S_i can be computed based on the deformation displacements at all anchor points and the zero displacement of the (equivalent) contact points of all the contact regions formed prior to S_i, using the interpolation method as described in Section V-E.

Once the global deformation is obtained, local-neighborhood deformation can be added inside the local neighborhood of S_i, as described in Section V-F. The final result is Shape$_i$.

The globally deformed shape of the elastic object as the result of all the m contacts is simply Shape$_m$.

Fig. 12(a) shows an example beam skeleton B_1 of contact S_1 built on an undeformed object, and Fig. 12(b) shows an example beam skeleton B_2 of a new contact S_2 built on an already deformed object due to the existing contact S_1.

VII. Implementation and Test Results

We have implemented the above method and applied it to real-time haptic rendering involving a virtual rigid body and an elastic object via a PHANToM Premium 1.5/6-DOF device,

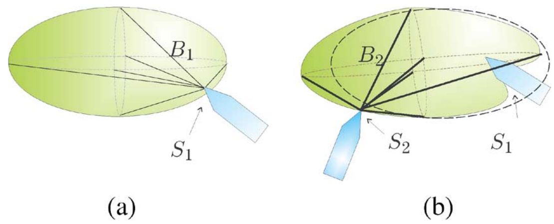


Fig. 12. Beam skeletons (in solid lines) on an elastic ellipsoid for (a) a contact S_1 on the undeformed object, (b) a new contact S_2 on the already deformed object due to the existing contact S_1.

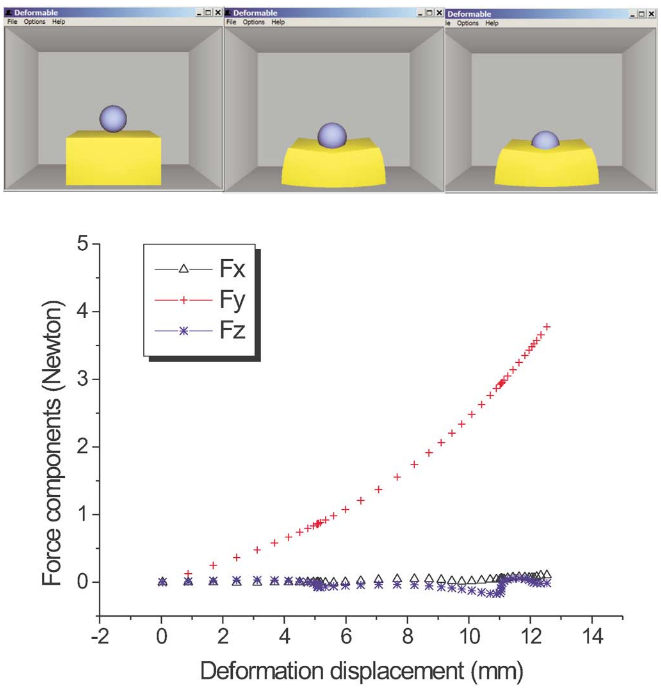


Fig. 13. Contact normal force rendering: downward motion and results.

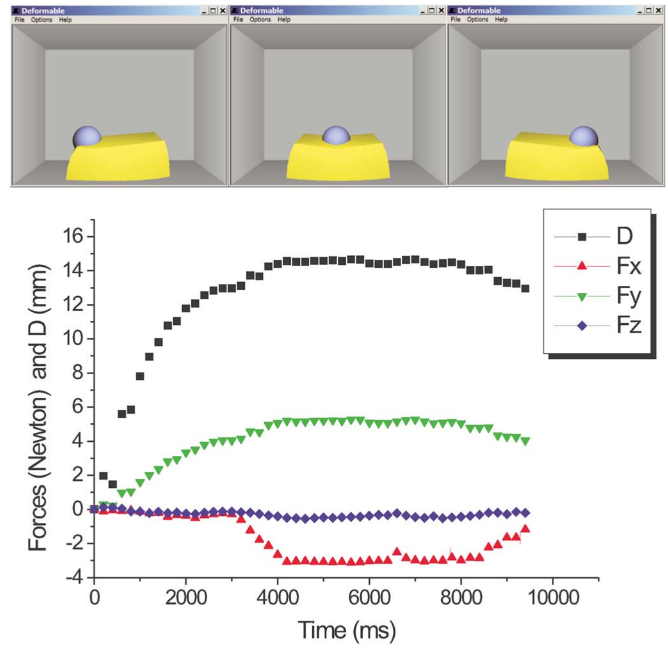


Fig. 14. Friction force rendering: compliant motion and results.

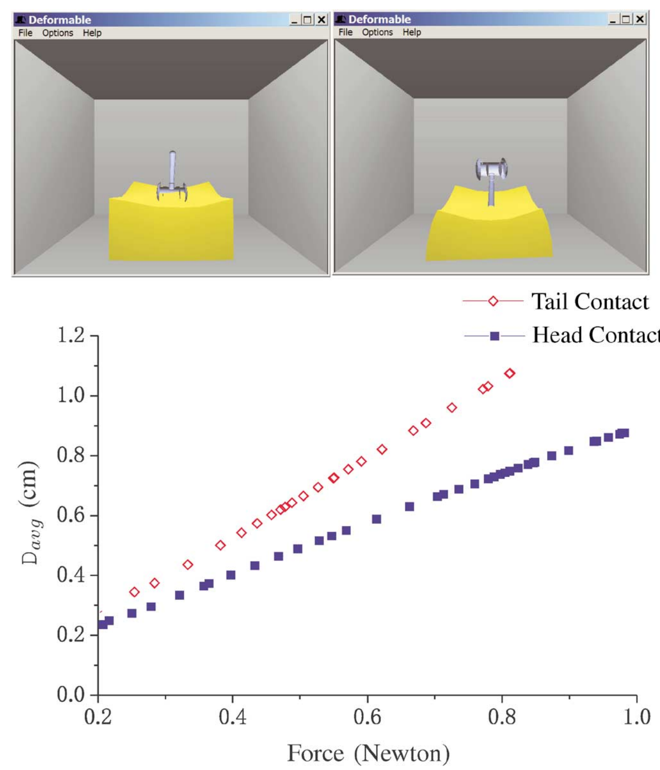


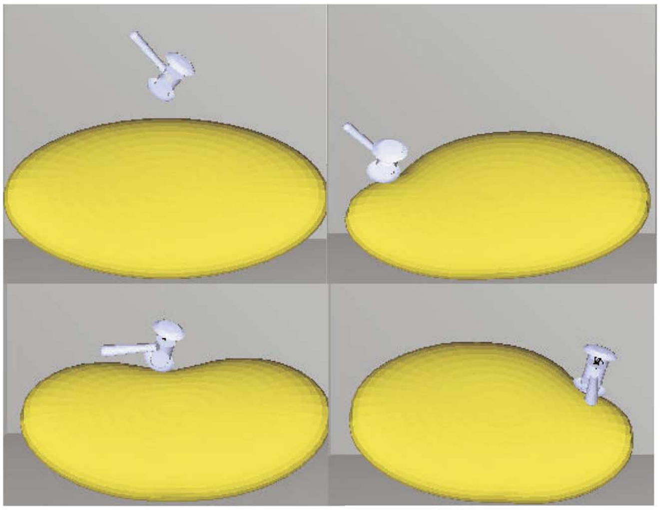
Fig. 15. Comparison of deformation in two different cases of single-region contact.

which is connected to a PC with dual Intel Xeon 2.4 GHz Processors and 1 GB system RAM. The human operator can virtually hold the rigid object A by attaching it to the haptic device and make arbitrary contact to the elastic object B (with its bottom center fixed, where a world coordinate system is set) by guarded motions, and perform compliant motions on the elastic object.

Figs. 13–18 show some test results, where the unit of force is Newton, the unit of length is mm, and the unit of time is ms. The world coordinate system in all examples is built as the following: x-axis points right; y-axis points up; and z-axis points out of the paper plane and is orthogonal to x and y axes following the right-hand rule.

Fig. 13 shows an experiment to test our method of nonlinear contact normal force rendering. The human operator virtually held the rigid ball and moved vertically down to the elastic cube. As expected, the normal support force increased nonlinearly as a function of the magnitude of deformation.

Fig. 14 shows an experiment to test contact friction force rendering. The human operator virtually held the rigid ball and moved vertically down to the elastic cube. When a contact was formed, the held ball was moved horizontally along the x-axis. As expected, the deformation amount along the y-axis increased from zero to an almost constant value, which generated a constant contact normal force F_y. The friction force was along the xz-plane, and had components F_x and F_z, with F_z almost zero. Initially, it was a static friction, and after the maximum static friction was reached and overcome, the friction became



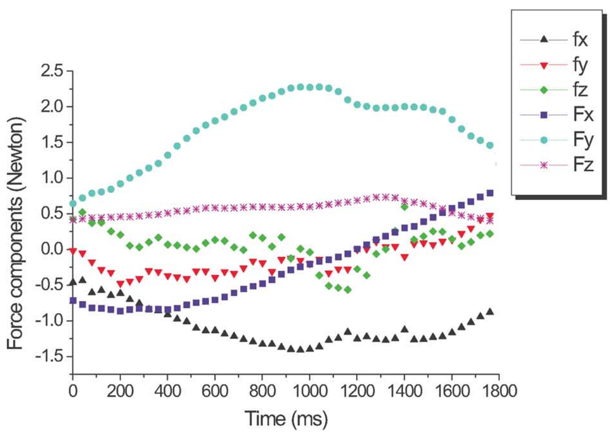


Fig. 16. Force rendering results when a mallet moved compliantly along an ellipsoid.

Fig. 17. Multiple contact rendering. (upper left) No contact. (upper right) One contact. (lower left) Two contacts. (lower right) One contact again.

dynamic with the direction always opposite the motion direction pointing approximately to the $-x$ direction.

Fig. 15 shows an experiment to compare the different effects of two cases with different contact areas on local neighborhood shape deformation. In case 1, a mallet's head contacted an

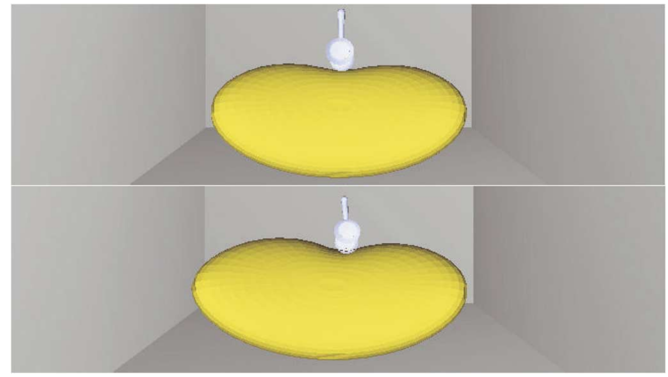


Fig. 18. Comparing shape deformations between homogeneous and nonhomogeneous objects. (top) A homogeneous rubber ellipsoid under a contact force. (bottom) A rubber ellipsoid with a smaller PVC ball inside its left side under the same contact force, which has a larger shape deformation on its left side.

elastic cube, and in case 2, the mallet's tail contacted the elastic cube. The contact area in case 1 was larger than the contact area in case 2. We can see that when the same force was applied, the average deformation displacement $\mathtt{D}_{\rm avg}$ of the contacting region in case 1 is smaller than that of case 2, which resulted in a shallower deformation that fits the related physics principle.

Fig. 16 shows a test example with a rigid mallet and an elastic ellipsoid. The human operator first moved down the mallet to make a contact with the elastic ellipsoid, and moved the mallet compliantly along the $+x$ direction, mostly, and slightly toward the $+z$ direction. Note that unlike in the case of a cube, where the contact normal force is along the y-axis, in this case, the contact normal generally has three components: F_x, F_y, and F_z. Here during the compliant motion, the value F_x changed from negative to positive, F_y was always positive and reached the maximum value when the mallet was moved to the upmost position of the ellipsoid, and F_z only had a small positive value, since the movement was slightly on the half of the ellipsoid toward the $+z$ direction. The friction force also has three components in general: f_x, f_y, and f_z. Here f_x always had a negative value, since the movement was toward the $+x$ direction, f_y increased from negative to positive, since the movement was first upward then downward along the y axis, and f_z was almost zero while the oscillation was due to the difficulty of moving the mallet in a straight motion.

Fig. 17 shows an example contact sequence that includes the shape of the deformable object under no contact, single contact, and two contacts. A rigid compass first touches the deformable object with one side pin and then both side pins. After that, the first pin breaks contact with the deformable object.

Fig. 18 shows a comparison of the shape deformations of a homogeneous object and a nonhomogeneous object. The homogeneous object is a rubber ellipsoid, and the nonhomogeneous object is a rubber ellipsoid with a smaller PVC ball inside its left side. We can see that asymmetric deformation is formed for the nonhomogeneous object.

In all of our experiments, modeling and computing haptic force per contact region took a constant and almost instant time of approximately 30 μs; that is, the computation had an update rate of approximately 33 kHz, regardless of the object's geometry. This was negligible compared with the time needed for

TABLE I
PARAMETERS USED IN EXPERIMENTS

General Parameters		**Rubber Parameters**		**PVC Parameters**	
Parameter	Value	Parameter	Value	Parameter	Value
$m(kg)$	1.0	$E(N/m^2)$	$3 \cdot 10^6$	$E(N/m^2)$	$33.77 \cdot 10^3$
$\beta_0^2\epsilon$	0.4	ν	0.5	ν	0.33
ω_0	0.2	$\rho(kg/m^2)$	1100	$\rho(kg/m^2)$	1050
K	10				
β	0.75				

real-time contact detection (Section III and Appendix I) plus shape updating for the elastic object, which was in the order of kHz. In all the examples, the numbers of triangles in the mesh models of objects range from 1000 to 3952.

Table I lists the parameters used in our experiments.

VIII. CONCLUSIONS

We have introduced a novel approach to model and render in real-time the nonlinear contact force response and the global and local shape deformation of a general elastic object caused by a rigid object contacting it and moving compliantly on it. Our approach can handle complex contact cases with multiple regions of contact. It achieves a good compromise of real-time efficiency and physical realism by taking advantage of nonlinear physics equations, elasticity principles, beam bending theory, and geometrical properties of general surfaces. We have implemented the approach to confirm its effectiveness. An update rate of over 1 kHz is achieved for the entire modeling and rendering process, including contact detection and rendering of both haptic force and graphic shape change. The next step in research is to add the effect of contact torque in such modeling and rendering. We also intend to further test and improve the implementation of the approach.

APPENDIX I
MULTIPLE CONTACT DETECTION

The collision-detection package SOLID [45] can detect the penetration distance between two objects in mesh models in real time, and also allow dynamic updates of meshes. These are properties useful to contact detection involving a deformable object. However, SOLID can only track one contact at each time step. In order to detect and track multiple contacts, we dynamically divide the deformable object into multiple parts and apply SOLID to the rigid object paired with each part of the deformable object, as detailed in the following.

When a contact point/region is formed, the mesh of the contact region and its local neighborhood on the deformable object are separated from the rest of the deformable object mesh. SOLID is applied to the divided submeshes separately to detect their contacts with the rigid object. Whenever a new contact region is formed, a new submesh of the contact region and its local neighborhood on the deformable object is separated. Whenever a contact no longer exists, the associated submesh is merged back to the deformable object mesh. In this way, multiple contact regions can be detected, tracked, and updated.

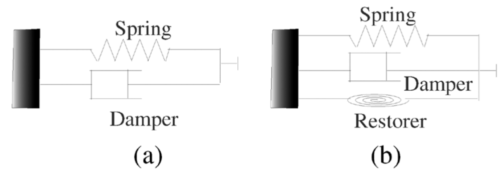


Fig. 19. Schematic of (a) a linear spring-damper model and (b) a nonlinear spring-damper-restorer model.

As long as the number of contact regions at any time is limited, e.g., below 10, our contact detection algorithm can achieve kHz update rate.

APPENDIX II
RELATION BETWEEN CONTACT FORCE AND DISPLACEMENT FROM THE DUFFING EQUATION

The Duffing equation essentially defines a nonlinear spring-damper-restorer model, as opposed to a linear spring-damper model, as shown in Fig. 19.

A general Duffing equation has the following form:

$$\ddot{x} + 2\mu\dot{x} + \omega_0^2 x + \epsilon\beta_0^2 x^3 = A cos \Omega t \tag{15}$$

where x is the deformation distance, $-\omega_0^2 x$ is the linear restoring term, $-2\mu\dot{x}$ is the damping term, $-\epsilon\beta_0^2 x^3$ is the nonlinear restoring term (with $|\epsilon| \ll 1$), and $A cos \Omega t$ is proportional to the external force and Ω is a constant. Note that the nonlinear restoring force item represents the nonlinear properties offered by the deformation of the nearby area. When $\epsilon > 0$, the nonlinear restoring force is greater than the linear restoring force, and it means hard nonlinear or hard spring. Otherwise, the nonlinear restoring force is smaller than the linear one, which means soft nonlinear or soft spring.

Now consider our problem of modeling the contact force response from the elastic object to the rigid object via the contact point in one time step. It is reasonable to assume that the external force exerted to the held rigid body from the human operator is constant during one short time step. Therefore, (15) becomes

$$\ddot{x} + 2\mu\dot{x} + \omega_0^2 x + \epsilon\beta_0^2 x^3 = A. \tag{16}$$

In (16), when $\epsilon \neq 0$, by using the equivalent frequency ω^* [34], $\omega^* = \omega_0 + 3\beta_0^2 a^2 \epsilon / 8\omega_0$, where a is the value of x at steady-state, we have

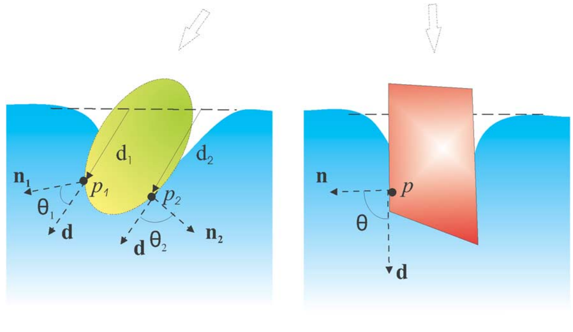
$$\ddot{x} + 2\mu\dot{x} + \omega^{*2} x = A$$

and its solution is

$$x = e^{-\mu t} C \sin \sqrt{\omega^{*2} - \mu^2} t + \frac{A}{\omega^{*2}}$$

where the first part in the above equation is the transient term, and the second part is the steady-state term. With the quasi-static assumption (Section II-C), under a large μ, the steady state is achieved when x reaches D at the end of one time step, i.e.,

$$D = a = \frac{A}{\omega^{*2}}. \tag{17}$$



(a) (b)

Fig. 20. Examples on contact force response. (a) Contact force response depends on both the depth at p_i along the direction of deformation and the angle difference between the contact surface normal at p_i and the deformation direction. (b) No contact force response at p when its normal direction is orthogonal to the deformation direction.

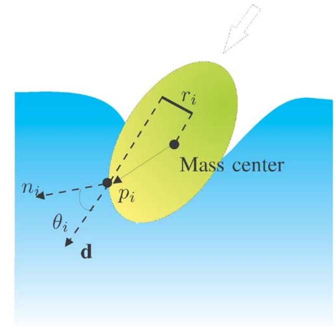


Fig. 21. Distribution of mass to point p_i.

That is, at $x = D = a$, the held rigid object becomes static, and the contact force response $\|\mathbf{F}_c\|$ from the elastic object balances the external force $\|\mathbf{F}_e\| = mA$, where m is the mass of the rigid object. Thus, from (17), by omitting the high-order term of ϵ (since $|\epsilon| \ll 1$), we have

$$\|\mathbf{F}_c\| = m\left|\omega_0^2 D + \frac{3\beta_0^2\epsilon}{4}D^3\right|.$$

Appendix III
Mass Load Distribution Over a Contact Region

To calculate the total contact force response to a single-region contact, we need to integrate the contact force responses over the contact region. Our implementation is to discretize force integration as the summation of contact forces at a number of evenly distributed contact points corresponding to different displacements of deformation (as shown in Fig. 5). We first need to determine those contact points and how to distribute the mass m of the held rigid object to them, before we can compute the contact force response to each such point p_i with displacement d_i based on the Duffing equation.

Our approach is to use the set of vertex points of the mesh model of the rigid object in the contact region, provided that these vertex points are evenly distributed. Let C indicate such a set of contact vertex points on the mesh.

At each point p_i in C, $i = 1, \ldots, k$, the amount of contact response force not only depends on the displacement d_i of p_i along the direction of deformation $\mathbf{d}$, but also depends on how the contact surface normal $\mathbf{n}_i$ at p_i relates to the deformation direction $\mathbf{d}$: the smaller the angle difference θ_i between n_i and d, the larger the contact response force to p_i (e.g., in Fig. 20(a), though $d_1 = d_2$, since $\theta_1 < \theta_2$, the contact force response to p_1 is greater than that to p_2). If n_i and d are parallel, the contact response force at p_i is the largest. On the other hand, if n_i and d are orthogonal, p_i does not contribute to the total contact response force [see Fig. 20(b)]. This fact is taken into account when we distribute the mass m to each p_i in the following, since the contact force at p_i is proportional to the mass contribution m_i at p_i according to the Duffing equation.

Based on the assumption that the external force from the user applies to the mass center of the held object, m_i can be approximated as inversely proportional to the projection r_i of the distance between p_i and the mass center on a plane whose normal is along the direction of deformation $\mathbf{d}$ (see Fig. 21) and weighted by $\cos\theta_i$

$$m_i = \frac{r_i\cos\theta_i}{\sum_{i=1}^{k} r_i\cos\theta_i}m.$$

The distribution of m over the contact region actually reflects the distribution of the pressure of the held rigid object on the contact region. Since such a pressure distribution depends on many specific factors, such as the specific shape of the held object, where the contact occurs, how large the deformation happens, the stress/strain properties of the elastic object, etc., it is difficult to obtain it exactly with a general method and in real time. The above formula for the distribution of m is a reasonable approximation.

With both d_i and m_i now known, the contact force response F_i at p_i can be computed from (1) (with d_i replacing D). Summing up all F_i gives the total contact force response F of the contact region.

Since it is difficult to compute the exact shape of the contact region in general, as such a region depends on the shape of the rigid object and how the contact is made, it is difficult to determine the exact set C of the contacting vertex points of the rigid object mesh in real time. Fortunately, the contributions of such mesh points inside the contact region can be very reasonably approximated by the contributions of the easily obtained set M of mesh points on the portion of the rigid object that is beneath the contact surface of the elastic object before this deformation where each point p_i in M satisfies $\mathbf{n}_i \cdot \mathbf{d}_i \geq 0$. Note that all contact cases can be categorized into two groups: $M > C$, Fig. 22(a), and $M = C$, Fig. 22(b). For the first group of contact, each noncontact point p_j in the set $M - C$ makes little contribution to the computation of the total contact force response F, since the angle θ_j between the surface normal at p_j and the direction of deformation d tends to be quite large, i.e., $\cos\theta_j$ tends to be close to zero, and the deformation displacement d_j is usually small.

Clearly, there is a tradeoff between the resolution of the mesh model (of the rigid object) and the computation efficiency of F. The finer the mesh model, the more accurate F is, but the slower the computation.

Appendix IV
Cross-Section Area of a Beam

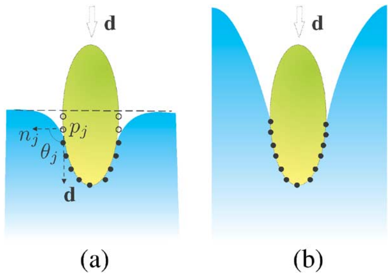


Fig. 22. Set M of points (points of both solid and hollow dots) versus the set C of points (points of solid dots). (a) $M > C$. (b) $M = C$.

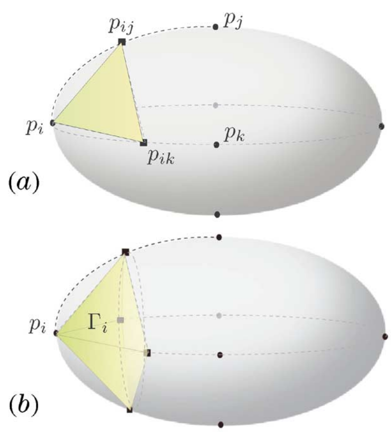


Fig. 23. Approximation of the Voronoi region of an anchor point p_i.

The surface of an elastic object can be partitioned into surface patches that form a Voronoi diagram based on the anchor points, such that each surface patch is a Voronoi region V_i of an anchor point p_i, which contains all the points on the elastic surface closer to p_i if traveling along the surface than to other anchor points.

Under a beam-skeleton B, we can compute the stresses and strains at each beam end p_i, which should represent the stresses and strains on the Voronoi region V_i of p_i. Therefore, we make the cross-section area s_i of a beam i proportional to the area of V_i.

Since B depends on the shape of the elastic surface before the current contact, which can be already deformed, it is difficult to find the area of each V_i precisely in real time. We have to use an approximation. Our strategy is to find all center points on the shortest curves connecting p_i to all adjacent anchor points.[2] Let p_{ij} be the center point between p_i and p_j. For two anchor points p_j and p_k adjacent to p_i, if p_j and p_k are also adjacent, then p_i, p_{ij}, and p_{ik} form a triangle, as shown in Fig. 23(a), where a segment of p_i's Voronoi region is approximated by the triangle bounded by p_i, p_{ij}, and p_{ik}. The area of V_i can be approximated as the sum of the areas of all such triangles, denoted as Γ_i, as shown in Fig. 23(b).

Let γ_i be the following:

$$\gamma_i = \frac{\Gamma_i}{\Gamma_1 + \ldots + \Gamma_n} \tag{18}$$

where n is the number of beams in the beam-skeleton B.

[2]We say two anchor points are adjacent if there exists a common boundary between their Voronoi regions.

The cross-section area of beam i in B should be proportional to γ_i, which captures the ratio of V_i's area relative to the total area of the elastic surface. In addition, since beam i is for computing the stress and strain at the anchor point p_i, which is at the center of the cross-section at the beam end, the cross-section area of beam i should be small enough so that the stress and strain at p_i can be approximated well. Therefore, we define s_i to be the cross-section area of beam i as follows:

$$s_i = \gamma_i s_c \tag{19}$$

where s_c is a constant area, and its value can be adjusted in order to best approximate the computation of stress and strain at an anchor point.

REFERENCES

[1] S. Gibson and B. Mirtich, "A survey of deformable modeling in computer graphics," Mitsubishi Electric Res. Lab., Cambridge, MA, Tech. Rep. TR-97-19, Nov. 1997.

[2] J. Lang, "Deformable model acquisition and validation," Ph.D. dissertation, Univ. British Columbia, Vancouver, BC, Canada, Oct. 2001.

[3] A. Al-Khalifah and D. Roberts, "Survey of modelling approaches for medical simulators," in *Proc. 5th Int. Conf. Disability, Virtual Reality, Assoc. Technol.*, Oxford, U.K., 2004, pp. 321–329.

[4] M. Lin, B. Baxter, M. Foskey, M. Otaguy, and V. Scheib, "Haptic interaction for creative processes with simulated media," in *Proc. IEEE Int. Conf. Robot. Autom.*, Washington, DC, 2002, pp. 598–604.

[5] C. Mendoza and C. Laugier, "Simulating cutting in surgery applications using haptics and finite element models," in *Proc. Virtual Reality Conf.*, 2003, pp. 295–296.

[6] D. d'Aulignac, R. Balaniuk, and C. Laugier, "A haptic interface for a virtual exam of the human thigh," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2000, vol. 3, pp. 2452–2457.

[7] S. Miyazaki, J. Hasegawa, T. Yasuda, and S. Yokoi, "A deformable object model for virtual manipulation based on maintaining local shapes," in *Proc. SCI*, Orlando, FL, Jul. 2001, vol. XI, pp. 100–105.

[8] J. Corso, J. Chhugani, and A. Okamura, "Interactive haptic rendering of deformable surfaces based on the medial axis transform," in *Proc. EUROHAPTICS*, Jul. 2002, pp. 92–98.

[9] 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.

[10] L. Nedel and D. Thalmann, "Real time muscle deformations using mass-spring systems," in *Proc. Comput. Graphics Int.*, 1998, pp. 156–166.

[11] S. Cotin, H. Delingette, and N. Ayache, "A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation," *Vis. Comput.*, vol. 16, no. 8, pp. 437–452, 2000.

[12] M. Bro-Nielsen, "Real-time simulation of deformable objects: Tools and application," in *Proc. Med. Meets Virtual Reality 5*, 1997, pp. 395–400.

[13] M. Mahvash and V. Hayward, "High-fidelity haptic synthesis of contact with deformable bodies," *IEEE Comput. Graphics Appl.*, vol. 24, pp. 48–55, Feb. 2004.

[14] Z. Cai, J. Dill, and S. Payandeh, "Toward deformation modeling with haptic feedback," in *Proc. IMECE*, Oct. 2000, pp. 1133–1138.

[15] C. Basdogan, C. Ho, and M. Srinivasan, "Virtual environments for medical training: Graphical and haptic simulation of laparoscopic common bile duct exploration," *IEEE/ASME Trans. Mechatron.*, vol. 6, no. 5, pp. 269–285, Sep. 2001.

[16] C. Duriez, C. Andriot, and A. Kheddar, "Interactive haptic for virtual prototyping of deformable objects: Snap-in tasks case," in *Proc. EUROHAPTICS*, 2003, pp. 159–175.

[17] O. Astley and V. Hayward, "Multirate haptic simulation achieved by coupling finite element meshes through Norton equivalents," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1998, pp. 989–994.

[18] J. Berkley, G. Turkiyyah, D. Berg, M. Ganter, and S. Weghorst, "Real-time finite element modeling for surgery simulation: An application to virtual suturing," *IEEE Trans. Vis. Comput. Graphics*, vol. 10, no. 3, pp. 314–325, May/Jun. 2004.

[19] S. Laycock and A. Day, "The haptic rendering of polygonal models involving deformable tools," in *Proc. EUROHAPTICS*, 2003, pp. 176–192.
[20] A. Frank, I. Twombly, T. Barth, and J. Smith, "Finite element methods for real-time haptic feedback of soft-tissue models in virtual reality simulators," in *Proc. IEEE Virtual Reality Conf.*, 2001, pp. 257–263.
[21] Y. Zhuang and J. Canny, "Haptic interaction with global deformations," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2000, pp. 2428–2433.
[22] B. Kim and J. Rossignac, "Localized bi-Laplacian solver on a triangle mesh and its applications GVU Tech. Rep. GIT-GVU-04-12, 2003.
[23] D. James and D. Pai, "A unified treatment of elastostatic contact simulation for real-time haptics," *Haptics-e*, vol. 2, no. 1, pp. 1–13, 2001.
[24] R. Balaniuk and I. Costa, "LEM—An approach for physically based soft tissue simulation suitable for haptic interaction," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2001, pp. 2337–2343.
[25] C. Mendoza and C. Laugier, "Realistic haptic rendering for highly deformable virtual objects," in *Proc. IEEE Conf. Virtual Reality*, Mar. 2001, pp. 264–269.
[26] K. Sundaraj, C. Mendoza, and C. Laugier, "A fast method to simulate virtual deformable objects with force feedback," in *Proc. ICARCV*, 2002, vol. 1, pp. 413–418.
[27] D. James and D. Pai, "Multiresolution greens function methods for interactive simulation of large-scale elastostatic objects," *ACM Trans. Graphics*, vol. 22, pp. 47–82, Jan. 2003.
[28] D. Aulignac, R. Balaniuk, and C. Laugier, "A haptic interface for a virtual exam of the human thigh," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, pp. 2452–2457.
[29] M. Bro-Nielsen, "Finite element modeling in surgery simulation," *Proc. IEEE*, vol. 86, no. 3, pp. 490–503, Mar. 1998.
[30] S. Cotin, H. Delingette, and N. Ayache, "Real-time elastic deformations of soft tissues for surgery simulation," *IEEE Trans. Visualiz. Comput. Graphics*, vol. 5, no. 1, pp. 62–73, Jan.–Mar. 1999.
[31] D. Nelson, D. Johnson, and E. Cohen, "Haptic rendering of surface-to-surface sculpted model interaction," in *Proc. Symp. Haptic Interfaces Virtual Environ. Teleoperator Syst.*, 1999, pp. 101–108.
[32] G. Debunne, M. Desbrun, M. Cani, and A. Barr, "Dynamic real-time deformations using space and time adaptive sampling," in *Proc. Comput. Graphics Interactive Tech., SIGGRAPH.*, 2001, pp. 31–36.
[33] X. Wu, M. Downes, T. Goktekin, and F. Tendick, "Adaptive nonlinear finite elements for deformable body simulation using dynamic progressive meshes," in *Proc. EUROGRAPHICS*, 2001, vol. 20, no. 3, pp. 349–358.
[34] S. Liu, *Nonlinear Equations in Physics*. Beijing, China: Peking Univ. Press, 2000.
[35] L. Miller, "Structural dynamics and resonance in plants with nonlinear stiffness," *J. Theor. Biol.*, vol. 234, pp. 512–524, 2005.
[36] Q. Luo and J. Xiao, "Modeling complex contacts involving deformable objects for haptic and graphic rendering," in *Proc. Robot.: Sci. Syst.*, Cambridge, MA, Jun. 2005, pp. 153–160.
[37] B. Berenberg, Dictionary of Composite Materials Technology. [Online]. Available: http://www.composite.about.com/library/glossary/bl-glossary.htm
[38] M. Teschner, S. Kimmerle, G. Zachmann, B. Heidelberger, L. Raghupathi, A. Fuhrmann, M.-P. Cani, F. Faure, N. Magnetat-Thalmann, and W. Strasser, "Collision detection for deformable objects," in *Proc. Computer Graphics Forum*, 2005, vol. 24, pp. 61–81.
[39] J. Mezger, S. Kimmerle, and O. Etzmuß, "Hierarchical techniques in collision detection for cloth animation," *J. WSCG*, vol. 11, no. 2, pp. 322–329, 2003.
[40] L. Raghupathi, L. Grisoni, F. Faure, D. Marchal, M. Cani, and C. Chaillou, "An intestine surgery simulator: Real-time collision processing and visualization," *IEEE Trans. Visualiz. Comput. Graphics*, vol. 10, no. 6, pp. 708–718, Nov./Dec. 2004.
[41] R. Bridson, S. Marino, and R. Fedkiw, "Simulation of clothing with folds and wrinkles," in *Proc. ACM/Eurographics Symp. Comput. Anim.*, 2003, pp. 28–36.
[42] A. Fuhrmann, G. Sobotka, and C. Gross, "Distance fields for rapid collision detection in physically based modeling," in *Proc. GraphiCon*, 2003, pp. 58–65.
[43] M. Teschner, B. Heidelberger, M. Mueller, D. Pomeranets, and M. Gross, "Optimized spatial hashing for collision detection of deformable objects," in *Proc. Vis., Modeling, Visualiz.*, 2003, pp. 47–54.
[44] B. Heidelberger, M. Teschner, and M. Gross, "Detection of collisions and self-collisions using image-space techniques," in *Proc. WSCG*, 2004, pp. 145–152.
[45] G. Bergen, "Efficient collision detection of complex deformable models using AABB trees," *J. Graphics Tools*, vol. 2, no. 4, pp. 1–13, 1997.
[46] I. Brown, "Abrasion and friction in parallel-lay rope terminations," Ph.D. dissertation, Univ. Cambridge, Cambridge, U.K., Mar. 1997.
[47] H. Howell and J. Mazur, "Amonton's law and fiber friction," *J. Textile Inst.*, vol. 44, no. 2, pp. 59–69, 1953.
[48] R. Temam and A. Miranville, *Mathematical Modeling in Continuum Mechanics*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
[49] Q. Luo, E. Staffetti, and J. Xiao, "Representation of contact states between curved objects," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 3589–3595.
[50] S. Tabachnikov, "The four-vertex theorem revisited—Two variations on the old theme," *Amer. Math. Monthly*, vol. 102, pp. 912–916, 1995.
[51] T. Belytschko, W. Liu, and B. Moran, *Nonlinear Finite Elements for Continua and Structures*. New York: Wiley, 2000.
[52] J. Smith and O. Sidebottom, *Elementary Mechanics of Deformable Bodies*. New York: Macmillan, 1969.
[53] A. Watt, *3D Computer Graphics*, 3rd ed. Reading, MA: Addison Wesley, 2000.
[54] R. Fenner, *Engineering Elasticity: Application of Numerical and Analytical Techniques*, ser. Mechanical Engineering. Chichester, U.K.: Ellis Horwood, 1986.

Qi Luo (S'05–M'06) received the Ph.D. degree in information technology from the University of North Carolina, Charlotte, in 2006. He also received the B.S. degree in physics from Nankai University, Nankai, China, and the M.S. degree in computer science from the University of North Carolina, Charlotte.

His research interests include haptic rendering, physics-based virtual reality, and robotics. He has served on program committees of the 2006 Robotics: Science and Systems Conference and the 2007 IEEE International Symposium on Assembly and Manufacturing.

Jing Xiao (S'88–M'90–SM'04) received the Ph.D. degree in computer, information, and control engineering from the University of Michigan, Ann Arbor, in 1990.

She is currently a Professor of Computer Science, University of North Carolina, Charlotte. From October to December 1997, she was a Visiting Researcher at the Scientific Research Laboratories, Ford Motor Company, and from January to June 1998, she was a Visiting Associate Professor at the Robotics Laboratory, Computer Science Department, Stanford University. From August 1998 to December 2000, she was the Program Director of the Robotics and Human Augmentation Program at the National Science Foundation. Her research interests cover robotics, haptics, and intelligent systems in general.

Dr. Xiao was an elected member of the Administrative Committee of the IEEE Robotics and Automation Society from 1999 to 2001. She is currently on the Membership Board of the IEEE Robotics and Automation Society and serves as the Associate Vice President for Membership Activities. She has been active in program committees of major IEEE robotics conferences for many years and has served in various roles including the Program Co-Chair of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, and the General Chair of the 2005 IEEE International Symposium on Assembly and Task Planning.