Shadow-Driven 4D Haptic Visualization

Hui Zhang and Andrew J. Hanson, *Member, IEEE*

Abstract—Just as we can work with two-dimensional floor plans to communicate 3D architectural design, we can exploit reduced-dimension shadows to manipulate the higher-dimensional objects generating the shadows. In particular, by taking advantage of physically reactive 3D shadow-space controllers, we can transform the task of interacting with 4D objects to a new level of physical reality. We begin with a teaching tool that uses 2D knot diagrams to manipulate the geometry of 3D mathematical knots via their projections; our unique 2D haptic interface allows the user to become familiar with sketching, editing, exploration, and manipulation of 3D knots rendered as projected images on a 2D shadow space. By combining graphics and collision-sensing haptics, we can enhance the 2D shadow-driven editing protocol to successfully leverage 2D pen-and-paper or blackboard skills. Building on the reduced-dimension 2D editing tool for manipulating 3D shapes, we develop the natural analogy to produce a reduced-dimension 3D tool for manipulating 4D shapes. By physically modeling the correct properties of 4D surfaces, their bending forces, and their collisions in the 3D haptic controller interface, we can support full-featured physical exploration of 4D mathematical objects in a manner that is otherwise far beyond the experience accessible to human beings. As far as we are aware, this paper reports the first interactive system with force-feedback that provides "4D haptic visualization" permitting the user to model and interact with 4D cloth-like objects.

Index Terms—knot theory, haptics, visualization

◆

1 INTRODUCTION

An early proponent of dimensional analogy was the 19th century psychologist and physiologist Gustave Fechner, who wrote a short story, *Space has Four Dimensions*, as part of his collection *Vier Paradoxe* published in 1846 under the pseudonym of Dr. Mises. Fechner described a two-dimensional creature, a "shadow man," projected to a flat screen. This shadow being could interact with other shadows, but could not conceive of a direction perpendicular to the screen. Interactive computer systems in fact work almost exclusively with shadows, i.e., representations of our 3D world cast upon 2D graphics screens by mathematical projection and rendering algorithms. Graphics methods allow us to add features to these shadows such as lighting, shading, and occlusion that we interpret via our learned perceptual models as being truly 3D, despite the fact that in truth their dimension is reduced.

Our task in this paper is to show how one can fully exploit projections to lower dimensions and the use of physically reactive projection-based haptic controllers to transform the task of interacting with the fourth dimension to a new level of "shadow-driven" physical reality. We like to think of this method intuitively as existing in a "shadow world," a term widely used in the classic literature, with clear interactive implications and an ancient context adopted by authors such as Fechner. Although it is possible to confuse what we call "shadow space" with the conventional illumination-based shadows of computer graphics technology, this will typically not be an issue in our treatment. We start from the fairly familiar idea of a knot "crossing diagram" drawing executed with a pen and paper, and extend that idea for pedagogical purposes to a haptic interface that is restricted to a plane, but still empowered to sketch, touch, and take control of a 3D mathematical knot through its projection to the controller plane. Having established the mechanisms and intuition of this artifice, we proceed to a full-featured extension to a 3D haptic system capable of manipulating 4D cloth-like surfaces with realistic physical characteristics, forces, and reactive collisions. In this way, we can proceed to attack families of significant problems in 4D intuitive visualization such as untying the apparently knotted twist-spun trefoil, canceling pinch-points in topological constructions, and modeling 4D "chain" structures consisting of linked spheres and ribbons.

2 RELATED WORK

The idea of treating shadow figures as a virtual reality goes back at least to Plato's *Allegory of the Cave* in the seventh book of *The Republic*. There the shadows are merely representations of objects seen by three-dimensional observers who are constrained so they cannot see the 3D physical world. Plato does not suggest that the shadows have the capability of interacting with one another, and this is the essential extension put forth by Fechner's work. The idea of cross-dimensional understanding has developed in many directions since then. Abbott's *Flatland* asked how two-dimensional creatures might attempt to understand three-dimensional space [1, 11], while Banchoff's pioneering work suggested how 3D computer-based projections could be used to study 4D objects [2, 3]. Other representative efforts include a variety of ways to render 4D objects (see, e.g., Noll [24], Hollasch [18], Banks [4], Roseman [30], and Egli, Petit, and Stewart [12]), and to extend lighting model techniques to 4D (see, e.g., [7, 16, 34]). Our own previous efforts suggested how haptic exploration of 4D objects can exploit topological continuity by ignoring illusory 3D surface intersections and focusing on the intrinsic 4D geometry [17].

In parallel to the above-cited work studying 4D mathematical objects via 3D projected images, there are also many approaches describing 3D curves and knots in terms of their shadows, largely motivated, one suspects, by the constraints of blackboard lectures and mouse-based computer interfaces. For example, sketching a knot with Scharein's Knotplot is done in a plane (basically the shadow world of the 3D embeddings) using a mouse-based control system [31], while Cohen et al. create 3D curves by correlating the curve with its sketched shadow to compute the curve's 3D shape [8].

Typical visualization methods for understanding higher dimensions employ a projection to 2D, or perhaps 3D, as a fundamental step; this helps the viewer to identify salient global features of the whole object and provides structural continuity when rotating the object's projection, or performing higher-dimensional rotations to change the projected image. Haptic exploration, being intrinsically limited to the physical world, also must project higher dimensional objects to three dimensions or less for interaction; within this context, knotted curves embedded in 3D can be projected to 2D and edited topologically within a consistent projected context to reveal complex topological relationships and structure. With the advent of modern interactive graphics technology we can begin to appreciate the challenge of sensing the phenomena from higher dimensions that baffled the two-dimensional shadow man of Fechner's story. In this paper we begin to answer the question: "How can we use touchable shadows and perceivable shadow-space forces to manipulate all the degrees of freedom of a four-dimensional world?"

- *Hui Zhang is with the Computer Science Department, Indiana University, Bloomington. Email: huizhang@cs.indiana.edu.*
- *Andrew J. Hanson is with the Computer Science Department, Indiana University, Bloomington. Email: hanson@cs.indiana.edu*

Manuscript received 31 March 2007; accepted 1 August 2007; posted online 27 October 2007. Published 14 September 2007.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

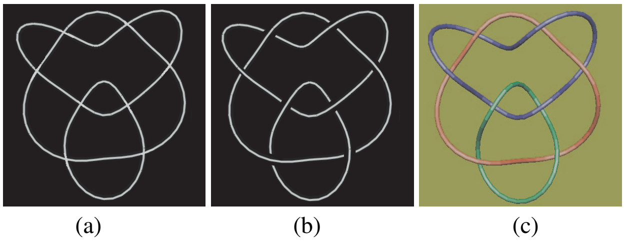

(a) (b) (c)

Fig. 1. (a) The 2D shadow diagram of this mathematical link has no 3D cues. (b) The 2-1/2 D knot diagram provides sufficient 3D depth information to characterize the 3D geometry. (c) Rendering with light and material adds apparent 3D geometry, depth, and shape to the 2D image.

3 OVERVIEW OF SHADOW MANIPULATION SCENARIO

People learn about the everyday world by combining sensory modalities. Knowledge of shape comes from a synthesis of the sensations of sight and touch acquired during exploration. By combining computer graphics with computer haptics, which endow the environment with intuitive force feedback, we can begin to create a sensory bridge from the shadow world to the higher-dimensional world we are trying to comprehend. While our ultimate challenge is to understand the physical 4D world using 3D haptics in its shadow space, it is very useful, perhaps even cognitively necessary, to introduce the basic features of our approach using a simplified analogy based on 2D shadows of 3D curves.

We therefore begin by examining the 2D projection of a 3D curve, noting specifically the overlapped neighborhoods that contain the 2D crossing points of pairs of 3D curve segments. If all we can see is the equivalent of pen strokes of a drawing on paper or the actual physical shadow of the 3D curve, we find the result in Figure 1(a), which is devoid of 3D information. This problem is typically overcome by employing the "crossing diagram" method illustrated in Figure 1(b); this corresponds essentially to a depth-buffered rendering with some embellishments to emphasize discontinuities in depth. By thickening the curve and providing geometry and material features combined with lighting and shading methods, we can get the additional improvement shown in Figure 1(c).

Now we can begin to see how to exploit a haptic probe in shadow space to facilitate full geometry manipulation: when the probe is constrained to the 2D shadow plane, the user can still freely edit the 3D structure in the *shadow-plane directions*. Permitting the user to modify the 3D depth of any point along a projection ray by using alternate states or modifiers completes the ability to support full-space geometric editing. We can thus, for example, use controls in the 2D shadow plane to supports 3D interactions that can create an arbitrarily complex topological knot. The key ideas of the overall scenario, exemplified by the 2D shadow world, can now be summarized as follows:

- *Create a shadow space in one lower dimension.* For example, take a 3D curve, create a 2D projection, and constrain the haptic probe's action to that 2D plane.
- *Add an extra $\frac{1}{2}$ dimension.* Display in the shadow space can be enhanced by showing occlusions of objects farther from the projection point when crossings occur. Similarly, sketching in the shadow space can query the user for an over-or-under choice when the curve-drawing shadow-space cursor collides with a part of the curve at the same depth in the real space.
- *Enhance the geometry and depth information in the higher dimension.* Computer graphics and visualization methods can enhance the perception of the geometry by color depth coding and exaggerated occlusion or crossing diagrams; rough sketches similarly benefit from global smoothing.
- *Make it physically touchable.* By modeling the physical interactions, collisions, forces, stretching, and momentum accurately in the full dimension, and projecting these touchable interactions down to the shadow space, we can support a substantial sensation of physical reality.
- *Manipulate projected images of objects embedded in the higher dimension.* We can combine direct shadow space controls with projection-ray controls to drag or deform object segments to arbitrary new places in the full-dimensional scene.
- *Explore the modeled objects with the haptic probe.* Pre-defined object models or newly constructed models can be explored by constraining the haptic probe to the model domain, with none of the problems of the real world, since the probe itself is not physical and is not bothered by illusory self-intersections of a continuous object in the shadow domain; furthermore, the probe itself does not collide with parts of the object, which is a major problem with touch-based exploration of objects such as real, physical knots [17].

In summary, by mapping the lower-dimensional user space (projection or shadow) to the full-dimensional object along either the dimensions of the projected space *or* along the projection rays, we can manipulate object deformations in the full space while experiencing physical artifacts such as force, inertia, momentum, and collisions induced by the higher dimensional simulation. In Figure 2, we show a very simple example with an "outside view" of a 3D curve deformation corresponding to standard knot move, as well as a ray-aligned move, controlled from the shadow space. Figure 3 illustrates the appearance of the entire user interface, including the virtual ray from the projected curve on the 2D screen to the user's 3D mental model of the artifact being manipulated.

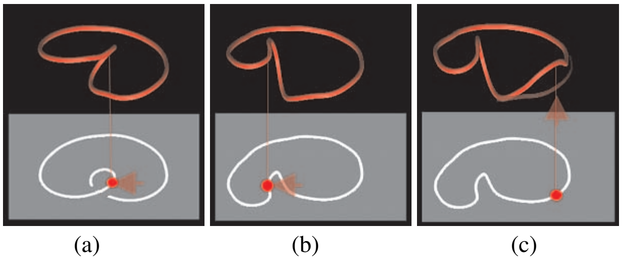

(a) (b) (c)

Fig. 2. (a) (Above) 3D object projected to 2D shadow (below). (b) Grabbing a point in the shadow and moving it laterally in the 3D space. (c) Grabbing a point and moving it along the shadow ray.

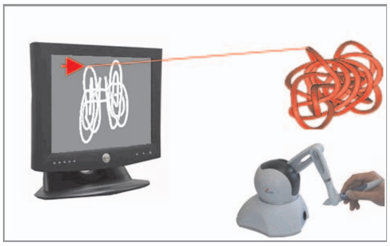


Fig. 3. Screen image, haptic probe, virtual ray from the physical space to the shadow, and the user's corresponding mental model during shadow-driven editing.

4 DETAILS OF IMPLEMENTATION MODELS

In this section, we describe the families of models used to implement the interaction procedures and user interfaces. We again focus on 2D projections of 3D curves to give explicit context, with the assumption that typical features will be straightforward to extend to 3D projections of 4D curves and surfaces; aspects requiring specific variants for the 3D/4D situation will be treated explicitly in a later section.

Our fundamental techniques are based in a wide variety of prior art, including haptic interfaces focusing on virtual realism (see, e.g., [5]), the exploration of unknown objects by robotic fingers (see, e.g., [25, 26], and other variants on haptic exploration techniques ([23] and [35]). Relevant methods of force feedback and user assistance include, e.g., the work of [36], [21], [13], [27], and [9]).

However, we have found many aspects of shadow-space manipulation to be unique, and thus we have been required to adopt customized hybrid approaches. For example, forces must be computed in the projection domain, but must maintain an accurate simulation of the higher dimensional physics to account for and create a haptic response to phenomena such as collisions and attendant over/under choices. Conversely, *apparent* collisions in the shadow of components that are physically separated in the physical dimensions must be ignored.

The basic modeling methods, components, and features characterizing our interface are summarized in the sections below.

4.1 Shadow Space Force Modeling

Our basic force model simulates a “sticky” stylus [17] in the shadow space using a damped spring configuration model; the probe can move freely in the shadow surface, but cannot move along a ray (which is technically possible in the 2D/3D situation, but impossible in 3D/4D).

The damped spring force model calculates the point $\mathbf{C}'$ as the projection of the haptic device proxy $\mathbf{C}$ on the shadow plane [32]. The difference $\mathbf{N} = \mathbf{C}' - \mathbf{C}$ is used to compute a generalized Hooke’s law force

$$\mathbf{F}_m = H|\mathbf{N}|^{1+\beta}\hat{\mathbf{N}} \ . \qquad (1)$$

Here H is a constant, and $\beta = 0$ for an ideal (linear) spring. This mechanical restoring force is applied whenever the stylus is displaced from the surface, but we allow force-free motion along the tangent direction to the surface to facilitate exploration of the surface structure. The user must apply substantial effort to overcome this force, so the stylus feels stuck to the surface. The damping force is taken to be

$$\mathbf{F}_d = -K_d\mathbf{V} \ , \qquad (2)$$

where $\mathbf{V}$ is the radial velocity used to smooth the force feedback.

4.2 Collision Avoidance and Repulsive Forces

Sketching Shadow Images. When sketching 2D shadows of 3D curves, collision detection is applied to detect whether the proxy apparently collides with other parts of the shadow image, and a repulsive force is rendered to prevent the haptic proxy from passing through segments in the shadow figure that actually collide in the full dimension; collisions are handled by making explicit over/under-crossing decisions.

Collision handling methods (see, e.g., [22] and [14], to mention only a few) detect a collision between virtual objects when they have just begun to penetrate each other. However, in a haptic interface, the colliding pair positions have physical manifestations, so one cannot simply shift both positions to undo the collision. Therefore we use a dynamic repulsive force to avoid collisions. The force model that we use to physically detect an impending collision and prepare for an over/under-crossing choice is

$$\mathbf{F}_r = -HS^{-1-\beta}\hat{\mathbf{V}} \ . \qquad (3)$$

Here S represents the distance between the probe itself and the impending collision with the shadow figure, and $\mathbf{V}$ is the radial velocity. This force slows down the haptic proxy’s velocity as it approaches an existing shadow image segment, thus allowing the system to detect and manage collisions in a physically realistic manner.

When a collision occurs between a piece of an edited object and an existing object in shadow space, users must make explicit over and under choices, e.g., by using modifier keys. The resulting displacement may be smoothed using the minimum distance energy method [19]. We thus have a “$2\frac{1}{2}$D” collision avoidance that effectively leverages skills developed from work with pen and paper, and exploits intuitive force feedback to aid the drawing process. This is closely related to the (non-haptic) shadow-driven methods for sketching and manipulating 3D curves advocated by Scharein [31] and by Cohen et al. [8].

Smoothing the Sketching Process. The $2\frac{1}{2}$D interface in principle is sufficient to allow the user to sketch a knot diagram on the given 2D shadow surface. However, in practice, this free-hand 2D constrained drawing introduces significant jitter (human and mechanical). To improve on this, we follow Haeberli’s *Dynadraw* method [15], connecting a virtual mass to the cursor position via a damped spring. As the user moves the cursor, the literal path is modified to create smooth, calligraphic strokes. From *Dynasculpt* (Snibbe [33]), a haptic variant of *Dynadraw*, we adopt the method of attaching a virtual mass-spring system to the haptic probe position to smooth the free-hand results.

Example: Creating a knot. In Figure 4, we illustrate typical steps for the shadow-space creation of a trefoil knot. Sample distances are interactively adapted, e.g., via bounding sphere checking, to make the final curve segments close to the same size and well-behaved (see [6]).

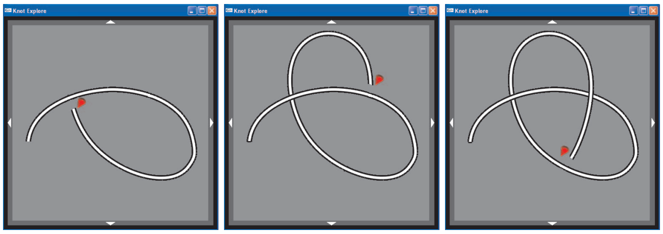


Fig. 4. Haptic knot creation via a sequences of under, over, under . . .

4.3 Multimodal Navigation

4.3.1 Haptic Feedback

The overall experience can be improved by constraining the probe to follow the local continuity of the object being explored. Tracing a real physical knot with one’s finger results in collisions of the rope with the fingertip, forbidding smooth navigation; the projected image of a knot can contain massive interruptions of visual continuity as well, as shown in Figure 5(a-b). The computer-based haptic interface, however, can do something real-life cannot do, which is to support a continuous motion that follows the continuity of the object being explored without encountering physical obstructions while overriding visual obstructions. The haptic navigation method resolves the apparent conflict between the continuous structure of the actual 3D knot and the visual discontinuities at occlusion boundaries in the 2D shadow domain [17], as shown in Figure 5(c).

Haptic navigation can be assisted by force suggestions that constrain the allowed motion, while assisting and guiding the user’s fingertip (the probe) towards the predicted position. The following steps describe our haptic servo loop model for adding force suggestions, as shown in Figure 5(d):

1. Get current haptic device coordinate $\mathbf{C}$, velocity $\mathbf{V}$, and the instantaneous update rate of the device R.
2. Compute the predicted haptic device coordinate $\mathbf{C_p} = \mathbf{C} + \mathbf{V} \cdot \frac{1}{R}$. ($\frac{1}{R}$ is the time step.)
3. Compute $\mathbf{C_p}'$ as projection of $\mathbf{C_p}$ on curve image $\mathbf{I}$.
4. Apply a damped spring force between $\mathbf{C}$ and $\mathbf{C_p}'$.

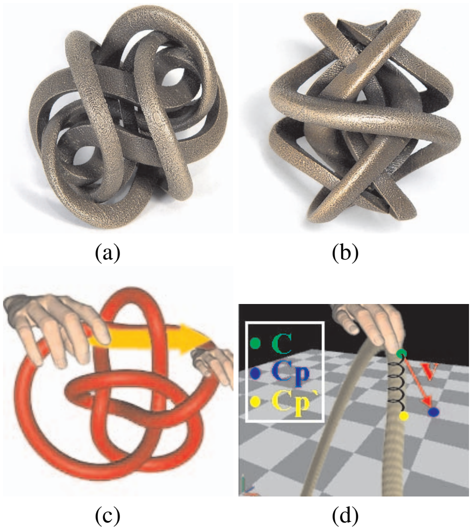


Fig. 5. (a-b) Tracing a real knotted rope is difficult, even when you are holding the physical object in your hands (`www.bathsheba.com/sculpt/clef/`). (c) A computer-based haptic probe supports a continuous unobstructed exploration of the *local continuity*. (d) Damped-spring forces can improve the exploration experience.

4.3.2 Auditory Feedback

Adding multimodal feedback can provide useful supplementary conceptual information. For example, if one explored a knotted curve with a probe constrained to the curve, but with no visual feedback, all knots would be the similar — just a path coming back to itself. In practice, knot structure is encoded by the location and character of the crossings (over and under crossings in the projection). We can add this information to either replace or redundantly supplement the visual display by adding an auditory signal that distinguishes each over and under crossing in the projection, as illustrated in Figure 6. Similar over and under crossings occur for surfaces projected from 4D to 3D, and these can be similarly signaled [17].

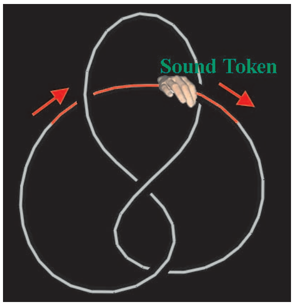


Fig. 6. Sound cues can be added to tell the user when crossings occur during exploration.

5 PHYSICAL INTERACTION WITH 3D SHADOWS OF 4D SURFACES

We now turn to our main objective, which is to create unique physical experience for the 4D world that can begin to make the strange more familiar.

The technique just described for controlling 3D curves projected to 2D has an exact analog that applies to projection from 4D to 3D. 4D shapes can be created, edited, and explored in their 3D projections by using the full three degrees of freedom of the haptic probe instead of artificially restricting ourselves to a 2D subspace. We typically want to investigate 4D surfaces instead of curves, since surfaces in 4D play many roles analogous to those of curves in 3D; in particular, spheres are the analogs of closed curves, and knotted curves are replaced by knotted spheres in 4D. Pushing and pulling on the projection of a 4D surface in the 3D projection manipulates its shape in all but the shadow-ray direction. By using alternate states or modifiers, the controller can activate shadow-ray-aligned motion, and for models or sketch results with very small shadow-ray components, we can handle collisions by forcing the modeled shape component to go "above" or "below" its colliding neighbor in the ray direction. Rigid 4D rotation controls can in turn expose any component in the 3D projection for selective editing.

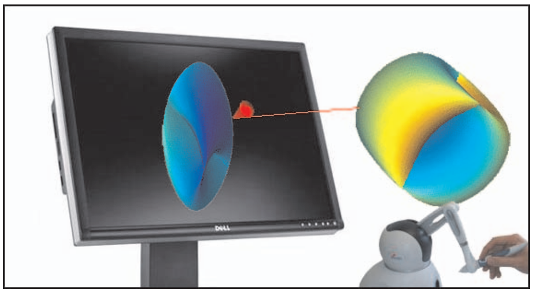


Fig. 7. Editing the 3D shadow image of 4D object embedded in a simulated 4D mathematical world.

Just as we can edit projections of 3D curves, we can analogously sketch the 3D shadows of 4D surfaces. We depend on the 3D collision mechanisms to locate possible crossings, and use a modifier to make over and under choices in 4D. Figure 7 shows the mental model of a user working with a 4D flat torus, while Figure 8 shows the first step of the construction of 4D "chain" consisting of linked spheres and circular ribbons.

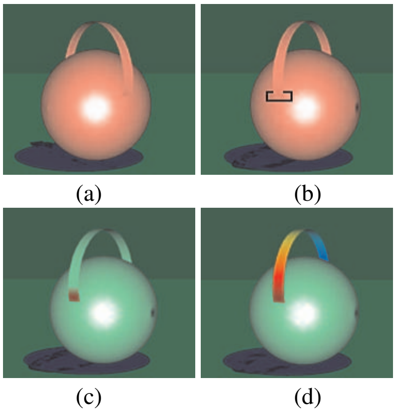


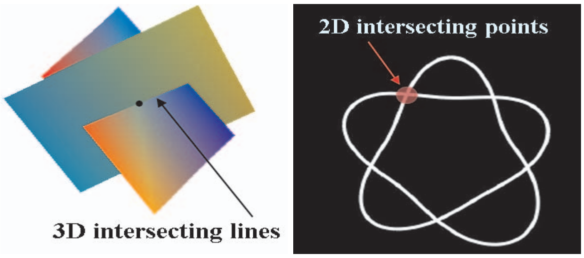
Fig. 8. (a) Two intersecting surfaces in 3D resulting from the projection of a 4D scene to its 3D shadow. (b) Crossing diagram of the surfaces; the front portion of the ribbon surface is closer to the 4D projection point than the spherical surface. (c) Above-below crossing markings using 4D depth coding. (d) Application of 4D depth smoothing to the ribbon.

5.1 Editing the 4D Embedding

The haptic device can freely edit a 4D shape (typically a surface) in the 3D shadow space using our basic control philosophy augmented by 4D physical simulation. The key to editing topological surfaces embedded in four dimensions is a proper collision handling mechanism that realizes material forces and prevents collisions from occurring in 4D.

5.1.1 Collision in Four Dimensions

To understand the nonintuitive mechanisms of 4D collision, let us start with a pair of two-dimensional planes through the origin in four-dimensional space (see Figure 9(a)). The two squares intersect in a single 4D point at the origin. In the three-dimensional projection, however, the planes appear to intersect along an entire line due to the projective collapse of the w dimension. When the surfaces are 4D depth color-coded, we can see that there is just one pair of points with the same fourth coordinate along the intersection line in the 3D projection. *Note that 4D collisions take place only if there is also a 3D collision in the shadow, but that 3D shadow collisions may not imply 4D collisions.*



(a) (b)

Fig. 9. (a) 4D Collision: surfaces are color-coded to show distance from the 4D projection point, so we see that only one point on the intersection line of the two planes has the same 4D coordinate. (b) A pair of 3D points that share same 2D projected coordinates.

We use Figure 9(b) to illustrate the algorithm for projection based collision detection: although 2D projections of 3D curves intersect with each other at the same projected coordinate, a 3D depth test must be performed on the pair of 2D intersecting points to determine if they actually collide in 3D. Similarly, a 4D depth collision test must be performed along the 3D lines where the projected images of the 4D surfaces intersect. The closest points on the 4D line segment pair are identified to compute the distance between them [37]. *4D collision occurs if, and only if, one or more pairs of points are located with the same fourth coordinate along the intersecting line.*

5.1.2 4D Collision Avoidance

Just as for 2D or 3D collision handling mechanisms, there are two possibilities for handling 4D collisions: resolve them after they happen and are detected using methods employed in [37], or never let them occur. Since the haptic probe on a colliding point pair is physical, one cannot simply shift both positions to undo the collision; thus, we typically choose to prevent 4D collisions before they happen, i.e., with 4D collision avoidance.

Let t_0 be an instant when there is no interpenetration between the two polygons embedded in 4D. Consider a time interval $[t_0, t_0 + \Delta t]$. Knowing the positions and velocities of each node of the 4D surface model at time t_0, it is possible to compute its positions at time $t_0 + \Delta t$ (we predefine a threshold value δ for the thickness of the 4D surface and any motion between two frames is clamped to be no greater than δ). Collision avoidance then consists of determining if one is heading towards a potential collision. Two cases occur:

- a vertex of a triangle in a 4D mesh is going towards a triangle of the mesh and the distance is less than δ (point-triangle collision)
- the edge of a triangle of a 4D mesh is going towards another triangle and their distance is less than δ (edge-edge collision)

If either of these two states exists, the pair of closest points on the colliding components are identified and l is defined as the 4D vector passing through them. Then an equal (but opposite) displacement along l is applied to each component along l, the displacement is just enough to take the component out of collision range. The basic 4D computations for point-triangle distance and edge-edge collision are extended from the 3D cases [28, 37], and in principle can be extended and applied to 2D surfaces embedded in arbitrary N-dimensional space.

5.1.3 4D Cloth-Environment Simulation

Visualizing 4D topological surfaces has long been a subject of fascination; however, most of the previous work has focused on correctly rendering images of such surfaces in the projected space. Four-dimensional mathematical and physical behavior (such as 4D collisions) cannot exist at all except in the "mind" of the computer's mathematical imagination. We are thus led to investigate an enhanced visualization environment where we model 4D surfaces with a 4D mass-spring system supporting physical interaction within the "shadow-driven editing" method.

4D mass-spring system. There have been a number of cloth models proposed in the last decade. Most of them aim to reproduce the physical behavior of cloth [10, 28], and others are able to generate approximate, but plausible animations in the context of interactive virtual reality gaming applications (see e.g., [20]). By extending the mass-spring system to four dimensions, we can in principle model the 4D mathematical and physical behavior of 4D topological surfaces. For the sake of generality, we will assume that each mass point i is linked to all the others with (linear) springs of rest length $l^0_{i,j}$ and stiffness $k_{i,j}$. This stiffness is set to zero if the actual model does not contain a spring between mass i and j. In the remainder of this paper, we will focus on 2-manifold deformable objects embedded in 4D. We will use the following three types of linear springs in our 4D interactive system:

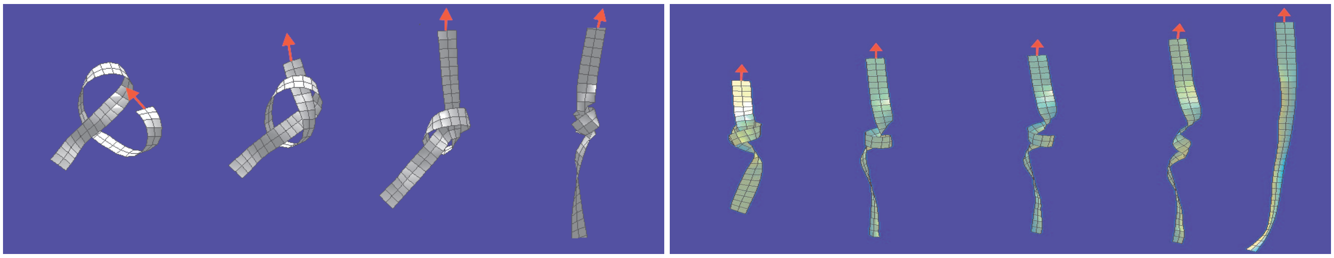
- springs linking masses $[i, j]$ and $[i+1, j]$, and masses $[i, j]$ and $[i, j+1]$, will be referred to as "structural springs";
- springs linking masses $[i, j]$ and $[i+1, j+1]$, and masses $[i+1, j]$ and $[i, j+1]$, will be referred to as "shear springs";
- springs linking masses $[i, j]$ and $[i+2, j+1]$, and masses $[i, j]$ and $[i, j+2]$, will be referred to as "bending springs";

Choosing the Shadow Plane. The orientation of the projection plane is an important factor in the interface. When sketching higher-dimensional objects in the projection, users typically will choose an informative projection to work on, much as we draw a knot diagram with pen and paper (see e.g., Figure 4 and Figure 8). In the editing interface, a separate mouse-driven orientation control is provided to adjust the 4D rotation matrix whose first two columns define the projection plane; this allows sufficient control to achieve a good projection for each particular editing subtask (see Figure 11).

Forces. The internal force $F_{int}[i, j]$ is the resultant of the tensions of the springs linking mass $[i, j]$ to its neighbors, where the calculation is based on Eq. 1 in four dimensions. The external force $F_{ext}[i, j]$ is applied when the mass $[i, j]$ is exposed to haptic manipulation. We note that our mass-spring system is configured in 4D, so the internal forces exerted by 4D springs are 4D vectors. The external forces applied by the 3D haptic device are still 3D vectors, and the 3D shadow space is the human-accessible dimension when interacting with 4D objects (this is the heart and soul of our "shadow editing" method). All these considerations allow us to compute the force $F_{i,j}(t)$ applied to the mass $[i, j]$ at any time.

Integration. The fundamental dynamical equation can be explicitly integrated across time by the Euler method:

$$\begin{cases} \alpha_{i,j}(t+\Delta t) = \frac{1}{m_{i,j}} F_{i,j}(t) \\ V_{i,j}(t+\Delta t) = V_{i,j}(t) + \Delta t \alpha_{i,j}(t+\Delta t) \\ P_{i,j}(t+\Delta t) = P_{i,j}(t) + \Delta t V_{i,j}(t+\Delta t) \end{cases} \tag{4}$$



(a) (b)

Fig. 10. (a) A knot tied in a 3D ribbon cannot be undone. (b) An apparent knot in the 3D projection of a 4D ribbon falls apart.

5.1.4 Haptic 4D Cloth Rendering

When we combine haptics with 4D cloth rendering, we support interaction with a 4D cloth simulation in the 3D shadow space, incorporating projected 4D collision forces. Selected 3D haptic cloth rendering methods [29] are adapted to our 4D environment. Our system uses constraint-based methods [38], haptically rendering a "rubber band line" from the haptic pointer to the picked vertex. Users can thus develop a correct interactive experience with the intuitive nature of unfamiliar 4D geometry by editing parts of the object in assorted 3D projections using 3D force feedback.

5.2 Examples

Interacting with 4D cloth-like objects using "shadow editing" methods can help users to develop a correct interactive experience with the intuitive nature of unfamiliar 4D geometry.

Visualizing 4D Collisions. One interesting 4D mathematical phenomenon is that in four dimensions loops and strings can always be untied although they appear knotted in the projected space (this can be confirmed by applying a rigid 4D rotation). With our accurate 4D mathematical and physical modeling and "shadow editing" method, users can interactively untie a 4D string that appears knotted (see Figure 10).

Tightening the 4D Spun Trefoil. In four dimensions, even though loops and strings can always be untied, surfaces can be knotted. One family of 4D knots consists of knotted spheres that are formed by spinning a knotted line segment in the fourth dimension to sweep out the surface; an example is the 4D spun trefoil knot. This 4D knotted sphere, just like a truly knotted 3D trefoil knot, cannot be untied. As illustrated in Figure 11(a-b), the user can try to pull apart the 4D knotted sphere (the 4D analog of a knotted piece of string). The effort fails, since in the rigorous computer simulation of the 4D physical world, the object is a true knot and cannot be untied by pulling on it. Neither can it be loosened, as illustrated in Figure 11(c-e). In this way, users can develop a correct interactive experience with the intuitive nature of unfamiliar 4D geometry.

Deforming the 4D Torus. The 4D embedded torus (the product of two circles, technically written as T^2) is an object of fundamental interest, with a standard model given by $\mathbf{x}(u,v) = (\cos u, \sin u, \cos v, \sin v)$. Although the orthogonal 3D graphics projection of this torus has two lines of self-intersection, the true surface is actually a smooth topological manifold in four dimensions. Our previous work presented a multimodal paradigm for exploring the smooth intrinsic features of topological surfaces like this embedded in four dimensions [17]. We can now obtain an alternative mental model of the structure of this surface by interactively deforming the object in the four dimensions. Figure 12 shows how to use local deformation to manipulate the projected image of the torus to actually eliminate the self-intersections in the 3D projection without altering or damaging the 4D surface in any significant way. Essentially, the pinch-points, where the X-shaped fold-over occurs, can be moved towards one another until they cancel both on the near and on the far side. The figure is color-coded for 4D depth, so the slight red bulge that remains in the middle is at a completely different depth from the blue surface; thus we can just push the red bulge through the blue surface in the 3D projection, yielding the final toroidal shape.

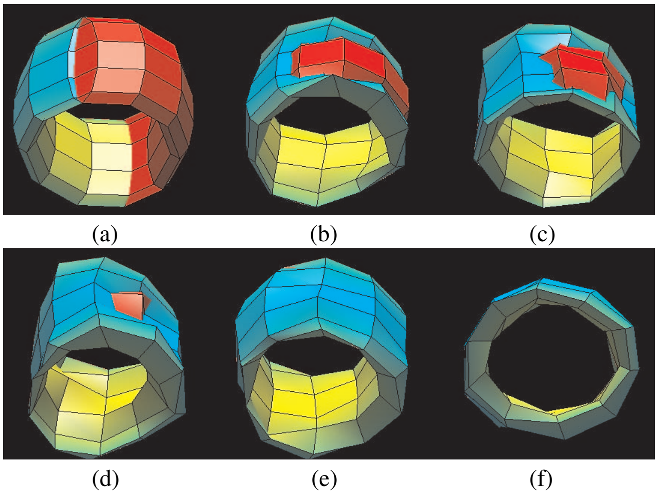


Fig. 12. (a)-(f): selectively applying local deformations to the 4D-embedded torus helps expose the underlying topological structure of the surface.

Physical Lifting in Four Dimensions. In Figure 13, we illustrate how materials interact in 4D by threading a ribbon into a piece of cloth, first going "above" in 4D, pulling under, and coming out the other side "below." If the cloth and ribbon are slippery, gravity will cause the cloth square to slide off the ribbon, much like two pieces of spaghetti in our 3D experience. In Figure 14, we perform the same operation except on a closed 2-sphere embedded in 4D; as long as we keep the two ends of the ribbon above the sphere, the captured point can no longer escape, and the sphere is lifted without being able to slide off; this is analogous to threading a string into a 3D ring. Just as we can link together rings in 3D to form a chain, we can in fact form 4D chains of alternating ribbons and spheres.

5.3 Multimodal Exploration of the Fourth Dimension

Haptic exploration of topological surfaces embedded in four dimensions can be supported by constraining the probe to conform to the local 4D continuity of the objects being examined [17]. The multimodal exploration in the 3D shadow image of the 4D embedded object passes continuously through the visually disruptive self-intersections of the 3D projection, revealing the full richness of the complex spatial relationships of the target shape.

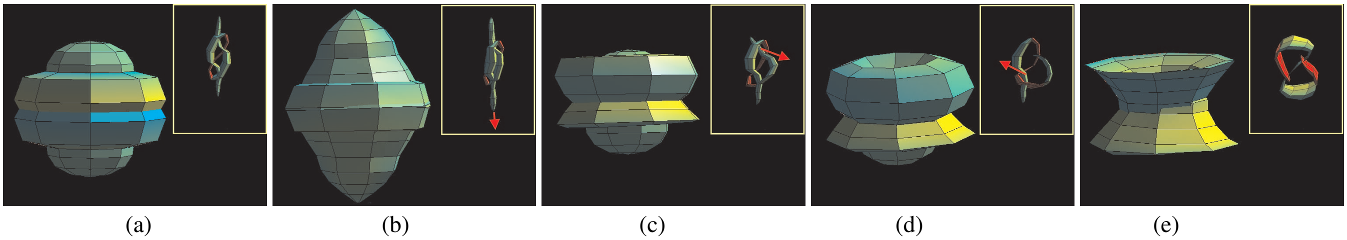

(a) (b) (c) (d) (e)

Fig. 11. (a,b) Tightening a 4D knotted sphere by pulling on the 3D shadow. (c,d,e) Pushing on the knotted sphere to loosen it.

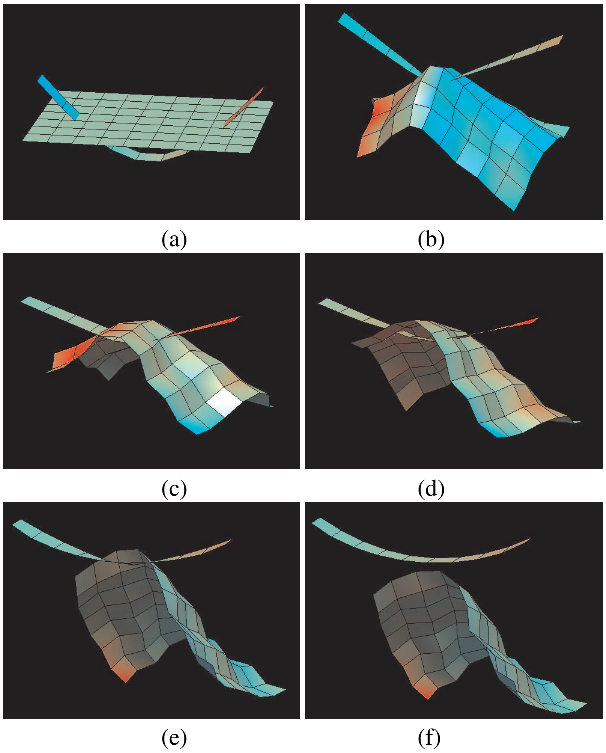

(a) (b) (c) (d) (e) (f)

Fig. 13. (a) Threading a 4D ribbon into a piece of 4D cloth (blue is under, red is over). (b,c) The 4D materials collide and exert lifting forces. (d,e,f) The 4D cloth slides off the ribbon due to gravity.

6 IMPLEMENTATION ENVIRONMENT AND PRELIMINARY USER STUDIES

Our implementation uses a SensAble Technology Omni PHANToM force-feedback haptic device combined with a high-performance graphics card supporting OpenGL. Our user interface is based on OpenGL and SensAble's OpenHaptics API. The software runs on a Dell PC desktop with 3.2GHz Intel Pentium 4 CPU. The haptic frame rate remains above 1000Hz for most tasks we have encountered (the haptic device requires a refresh rate of about 1000Hz in order to give the kinesthetic sense of stiff contact).

The 2D/3D system matches the shadow plane with the 2D computer screen, and a 2D cursor represents the haptic proxy in the scene. Haptic functions such as sketching, editing, and exploration of 3D curves projected to the 2D plane are enhanced by real-time force-feedback. The 2D/3D system has been successfully used to manipulate a variety of knots; an example manipulation is shown in Figure 15. The haptic control of 2D projected images of 3D knots is smooth and natural, and the user senses the collisions in the higher dimension when taking control of the shadow figures.

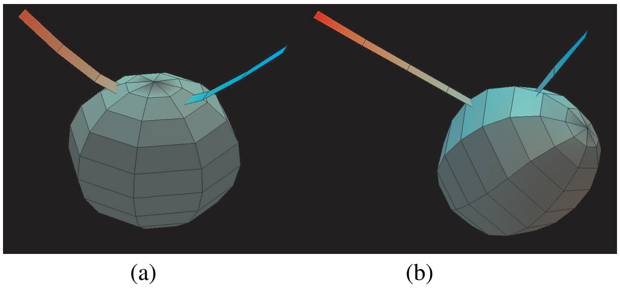

(a) (b)

Fig. 14. (a) Threading a piece of 4D ribbon into a closed 4D-embedded 2-sphere. (b) When lifted against gravity, the 4D linking cannot fall off.

Eight students from a topology class used the multimodal exploration tool for 3D mathematical knots before they were exposed to knot theory in the classroom. Haptic navigation on both 2D knot diagrams (see Figure 1(b)) and 3D rendered images (see Figure 1(c)) were supported to assist the participants' understanding of the relation between 2D knot diagrams and the actual 3D mathematical knot being represented. A set of 10 questions was used in our questionnaire. These questions focus on different aspects such as participants' acceptance of the "shadow-driven" exploration tool, haptic methods for aiding understanding of 3D knot structure, and auditory cues as supplementary information (measured using a 5-point Likert scale). Tasks such as determining whether two given knot diagrams represent the same knot and how to convert a true knot into an unknot by changing an individual crossing from "over" to "under" are also included in the questionnaire. Seven students (87.5%) were able to finish all the tasks in the questionnaire. Their response to the system was quite positive, as shown in Table 1 (scale 1 corresponds to "strongly disagree," 5 corresponds to "strongly agree"). Suggestions for improving the user interface were solicited; for example, students suggested constructing the Gauss code of the knot being explored in parallel with the crossing-triggered sound cues. We are incorporating some of this useful feedback into our 2D/3D and 3D/4D interfaces. Further quantitative analyses of these results, combined with studies of a select number of more advanced subjects using the 3D/4D interfaces, are in progress and will be reported in appropriate venues.

Table 1. Participants' evaluation of knot exploration system

	Show Driven Exploration Tool	**Haptic Method**	**Auditory Cues**
Evaluation (1 to 5)	M=4.0 SD=1.44	M=3.9 SD=1.25	M=3.7 SD=1.03

7 CONCLUSION AND FUTURE WORK

We have discussed a family of haptic methods for intuitively controlling the representations of higher-dimensional phenomena in their shadow spaces. By adding force-feedback to the concept of reduced-dimension object manipulation controllers, we have created a unique way of building intuition about 3D knots and curves in a 2D touchable space; this approach is naturally extensible to a full 3D haptic con-

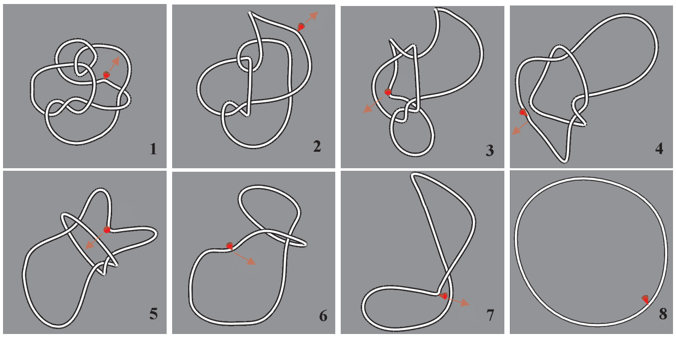


Fig. 15. Selectively applying local deformations to the 3D curves exposes the starting object as isotopic to the unknot.

troller that provides touchable intuition and force feedback through simulated representations of shapes in a 4D world. Starting from this basic framework, we plan to proceed to attack families of significant problems in 4D intuitive visualization such as the interactive manipulation of apparently knotted, but actually unknotted, spheres in 4D. Other planned future work will extend the range of objects for which we can support 4D physical modeling to include more complex knots, links, and Riemann surfaces, as well possibly supporting three-manifolds in addition to curves and surfaces.

ACKNOWLEDGMENTS

This work was supported in part by National Science Foundation grants IIS-0430730 and CCF-0204112. We are grateful to Charles Livingston for his generous assistance and advice.

REFERENCES

[1] E. A. Abbott. *Flatland*. Dover Publications, Inc., 1952.

[2] T. F. Banchoff. Visualizing two-dimensional phenomena in four-dimensional space: A computer graphics approach. In E. Wegman and D. Priest, editors, *Statistical Image Processing and Computer Graphics*, pages 187–202. Marcel Dekker, Inc., New York, 1986.

[3] T. F. Banchoff. Beyond the third dimension: Geometry, computer graphics, and higher dimensions. *Scientific American Library*, 1990.

[4] D. Banks. Interactive display and manipulation of two-dimensional surfaces in four dimensional space. In *Symposium on Interactive 3D Graphics*, pages 197–207. ACM, 1992.

[5] W. V. Baxter, V. Scheib, and M. C. Lin. dAb: Interactive haptic painting with 3D virtual brushes. In E. Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, pages 461–468. ACM SIGGRAPH, ACM, 2001.

[6] J. Brown, J. C. Latombe, and K. Montgomery. Real-time knot-tying simulation. *The Visual Computer*, 20(2-3):165–179, 2004.

[7] S. A. Carey, R. P. Burton, and D. M. Campbell. Shades of a higher dimension. *Computer Graphics World*, pages 93–94, October 1987.

[8] J. Cohen, L.Markosian, R. Zeleznik, J. Hughes, and R.Barzel. An interface for sketching 3d curves. In *SI3D '99: Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 17–21, New York, NY, USA, 1999. ACM Press.

[9] C.W.Reynolds. Steering behaviors for autonomous characters. In *The proceedings of the 1999 Game Developers Conference*, pages 763–782, 1999.

[10] M. Desbrun, M. Meyer, and A. H. Barr. Interactive animation of cloth-like objects for virtual reality. In *Cloth modeling and animation*, pages 219–239, Natick, MA, USA, 2000. A. K. Peters, Ltd.

[11] A. K. Dewdney. *The Planiverse: Computer Contact with a Two-Dimensional World*. Poseidon Press, 1984.

[12] R. Egli, C. Petit, and N. F. Stewart. Moving coordinate frames for representation and visualization in four dimensions. *Computers and Graphics*, 20(6):905–919, November–December 1996.

[13] B. Forsyth. Intelligent support of interactive manual control: Design, implementation and evaluation of look-ahead haptic guidance. Master's thesis, The University of British Columbia, 2004.

[14] S. Gottschalk, M. C. Lin, and D. Manocha. OBBTree: A hierarchical structure for rapid interference detection. *Computer Graphics*, 30(Annual Conference Series):171–180, 1996.

[15] P. Haeberli. Dynadraw. *Grafica OBSCURA*, July 1989.

[16] A. J. Hanson and P. A. Heng. Illuminating the fourth dimension. *Computer Graphics and Applications*, 12(4):54–62, July 1992.

[17] A. J. Hanson and H. Zhang. Multimodal exploration of the fourth dimension. In *Proceedings of IEEE Visualization*, pages 263–270, 2005.

[18] S. Hollasch. Four-space visualization of 4D objects. Master's thesis, Arizona State University, 1991.

[19] M. Huang, R. P. Grzeszczuk, and L. H. Kauffman. Untangling knots by stochastic energy optimization. In *VIS '96: Proceedings of the 7th conference on Visualization '96*, pages 279–ff., Los Alamitos, CA, USA, 1996. IEEE Computer Society Press.

[20] J.Lander. Devil in blue faceted dress: Real-time cloth animation. *Game Developer*, 1999.

[21] J. M. Kennedy. Optics and haptics: The picture. In *The conference on Multimodality of Human Communication: Theory, Problems and Applications.* University of Toronto, 2002.

[22] T. Larsson and T. Akenine-Möller. Collision detection for continuously deforming bodies. In *Eurographics 2001*, pages 325–333, Manchester, September 2001. Eurographics Association.

[23] L.Kim, G. Sukhatme, and M.Desbrun. A haptic-rendering technique based on hybrid surface representation. *IEEE Comput. Graph. Appl.*, 24(2):66–75, 2004.

[24] M. A. Noll. A computer technique for displaying n-dimensional hyperobjects. *Communications of the ACM*, 10(8):469–473, August 1967.

[25] A. M. Okamura. *Haptic Exploration of Unknown Objects*. PhD thesis, Stanford University, Department of Mechanical Engineering, California, USA, June 2000.

[26] A. M. Okamura and M. R. Cutkosky. Feature detection for haptic exploration with robotic fingers. *The International Journal of Robotics Research*, 20(12):925–938, Dec. 2001.

[27] J. G. Park and G. Niemeyer. Haptic rendering with predictive representation of local geometry. In *HAPTICS*, pages 331–338, 2004.

[28] X. Provot. Collision and self-collision handling in cloth model dedicated to design garments. In *Proc. Computer Animation and Simulation'97*, pages 177–189, 1997.

[29] C. Raymaekers, L. Vanacken, E. Cuppens, and K. Coninx. A comparison of different techniques for haptic cloth rendering. In *HAPTEX '05: Proceedings of the VR Workshop on Haptic and Tactile Perception of Deformable Objects*, pages 56–63. Magnenat-Thalmann N.[edit.], e.a., s.l., 2005.

[30] D. Roseman. Twisting and turning in four dimensions, 1993. Video animation, Department of Mathematics, University of Iowa, and the Geometry Center.

[31] R. G. Scharein. *Interactive Topological Drawing*. PhD thesis, Department of Computer Science, The University of British Columbia, 1998.

[32] SensAble, Inc. *3D Touch SDK OpenHaptics Toolkit Programmer's Guide*, 2004.

[33] S. Snibbe, S. Anderson, and B. Verplank. Springs and constraints for 3d drawing. In *Proceedings of the Third Phantom Users Group Workshop*, Dedham, MA, 1998.

[34] K. V. Steiner and R. P. Burton. Hidden volumes: The 4th dimension. *Computer Graphics World*, pages 71–74, February 1987.

[35] W. Yu, R. Ramloll, and S. Brewster. Haptic graphs for blind computer users. In *First International Workshop on Haptic Human-Computer Interaction*, pages 102–107. British HCI Group, Springer, 2000.

[36] M. A. Zahariev and C. L. MacKenzie. Auditory, graphical and haptic contact cues for a reach, grasp, and place task in an augmented environment. In *Proc. of 5th Intl. Conf on Multimodal Interfaces (ICMI)*, pages 273–276, New York, 2003. ACM, ACM Press.

[37] H. Zhang and A. J. Hanson. Physically interacting with four dimensions. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. V. Nefian, M. Gopi, V. Pascucci, J. Zara, J. Molineros, H. Theisel, and T. Malzbender, editors, *ISVC (1)*, volume 4291 of *Lecture Notes in Computer Science*, pages 232–242. Springer, 2006.

[38] C. B. Zilles and J. K. Salisbury. A constraint-based god-object method for haptic display. In *IROS '95: Proceedings of the International Conference on Intelligent Robots and Systems-Volume 3*, page 3146, Washington, DC, USA, 1995. IEEE Computer Society.