

K. E. Jordan <sup>1</sup>, Lance E. Miller <sup>2</sup>, E. L. F. Moore <sup>3</sup>,  
T. J. Peters <sup>4</sup>, A. C. Russell <sup>5</sup>

## MODELING TIME AND TOPOLOGY FOR ANIMATION AND VISUALIZATION

ABSTRACT. The art of animation relies upon modeling objects that change over time. A sequence of static images is displayed to produce an illusion of motion, which is frequently trusted to be topologically meaningful. A careful analysis exposes that formal topological guarantees are often lacking. This lack of formal justification can lead to subtle, but significant, flaws regarding topological integrity. A modified approach is proposed that integrates topological rigor with a continuous model of time. Examples will be given for splines widely used in many applications, with particular emphasis upon scientific visualization for molecular modeling. Moreover, the approach of choosing a family of functions and studying their topological properties over time should be broadly applicable to other domains. Prototype animations are available for viewing over the web.

**Keywords:** Ambient isotopy; computational topology; temporal aliasing; animation; visualization; curve approximation.

### 1. INTRODUCTION: ANALYSIS OF TOPOLOGICAL FLAWS IN TIME MODELS

The animation paradigm described in the abstract was established by human animators and persists in contemporary computer animation. In computer animation, the individual static images are known as *frames* and distinct geometric models are created for each frame. Each geometric model is then displayed by a supporting computer graphics system. The creation of the successive geometric models typically proceeds by edits to some earlier static geometry. It is possible that a well-intentioned edit could introduce a geometric self-intersection. This could produce unwanted artifacts, as the geometric models are often assumed to have no self-intersections over

---

<sup>1</sup>IBM Corporation, One Rogers St., Cambridge, MA 02142

<sup>2</sup>Department of Computer Science, University of Connecticut, Storrs, CT 06269-3009 USA. Partial funding for Lance E. Miller was from NSF grant CCF 0429477 and from a 2006 IBM Doctoral Fellowship. All statements in this publication are the responsibility of the authors, *not* of these funding sources.

<sup>3</sup>Department of Computer Science, University of Connecticut, Storrs, CT 06269-3009 USA. Partial funding for E. L. F. Moore was from NSF grant CCF 0429477.

<sup>4</sup>Department of Computer Science and Engineering and Department of Mathematics, University of Connecticut, Storrs, CT 06269-3155 USA, tpeters@cse.uconn.edu, Partial funding for T.J. Peters was from NSF grant CCF 0429477 and from a 2005 IBM Faculty Award.

<sup>5</sup>Department of Computer Science and Engineering University of Connecticut, Storrs, CT 06269-3155 USA, acr@cse.uconn.edu. Partial funding for A.C. Russell was from NSF grants CCF 0429477 and CCR 0226504.

*Date:* 28th April 2007.

all frames because graphic subsystems can have difficulty in properly displaying self-intersecting objects.

For example, in creating a protein-enzyme simulation, the actual dynamics of the underlying physical system may naturally avoid topological changes (such as fracture of a twisting DNA strand), but the underlying algorithms are not designed to ensure the global topology of the perturbing model. As the objects can be geometrically complex, it becomes impractical and unreliable for humans to attempt to visually detect any unwanted artifacts. The dominant approach today is to post-process the geometry in each frame by testing for self-intersections as opposed to modeling the continuous motion [22].

A careful conceptual analysis of this approach exposes two main difficulties:

- (1) The occurrence of self-intersections between frames would remain undetected.
- (2) Algorithms for detecting geometric self-intersections are compute intensive.

The first difficulty is inherent in the approach of only inspecting static images. That approach tacitly assumes that the time differences between each frame are sufficiently small to preclude self-intersections between frames. However, such self-intersections can occur and disappear instantaneously, as illustrated in Figure 1, below. The drawings of Figure 1 have a simplified schematic of an instantaneous, pernicious topological change. The topology of unknot (on the left) changes *instantaneous* upon the appearance of the self-intersection shown at the top of the middle image. Subsequently, further topological change results in a trefoil knot.

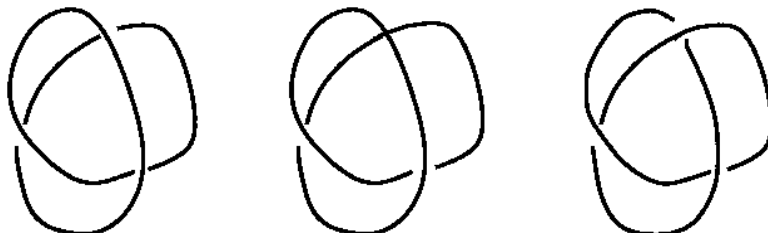


FIGURE 1. Self-intersection Precedes Knot Change

The topological emphasis presented here will prevent even these instantaneous temporal artifacts. This new computational approach to modeling time for animation and dynamic visualization relies upon a lemma proven here to permit the synthesis of several results into a new methodology. Brief mention will be made regarding performance benefits relative to the second stated difficulty.

## 2. RELATED WORK

The proposed approach focuses upon topological equivalence over the time variable used in simulations. Preventing the appearance of new self-intersections has been important in animation [14] and motivated the extensions developed here. Progress in

computational topology invariants has advanced greatly through research on biological applications [2, 3, 11, 12]. The role of knot characteristics has been prominent for molecular modeling [35]. The classical tubular neighborhoods are fundamental for isotopy [19] equivalence, as invoked in a contemporary algorithm for a piecewise linear (PL) ambient isotopic approximant to a parametric curve [23].

A seminal work on the integration of graphics for simulation [36] emphasized efficient algorithms for culling, scaling and clipping of geometry when no topological changes were expected. A focus on polylines or polyhedra [10, 13, 15, 18] or other specialized shapes [1, 32] has been motivated by performance concerns. Visualization and graphics of massive data sets have been studied [1, 9, 16, 17, 18, 20] as well as applications for the life sciences [21, 34].

The current authors have contributed to the emphasis upon algorithms for isotopic equivalence between a parametric curve and its piecewise linear (PL) approximant [24, 26, 28, 29]. Algorithms for preservation of topological form during perturbation of vertices and control points, for polyhedra and splines, respectively, have given practically computable limits on the distance these points can be perturbed [5, 7, 6].

### 3. ISOTOPY FOR BÉZIER CURVES

A curve will be the image of a continuous function  $\mathbf{c} : [0, 1] \rightarrow \mathbb{R}^3$ . These curves are necessarily compact. These curves can be open or closed and have self-intersections. However, it is noted that any two compact open curves which do not self-intersect are ambient isotopic. Since the focus will be on ambient isotopic equivalence, that definition is given.

**Definition 3.1.** *Let  $X$  and  $Y$  be two subspaces of  $\mathbb{R}^n$ . A continuous function*

$$H : \mathbb{R}^n \times [0, 1] \rightarrow \mathbb{R}^n$$

*is an **ambient isotopy** between  $X$  and  $Y$  if  $H$  satisfies the following conditions:*

- (1)  $H(\cdot, 0)$  is the identity,
- (2)  $H(X, 1) = Y$ , and
- (3)  $\forall t \in [0, 1], H(\cdot, t)$  is a homeomorphism from  $\mathbb{R}^n$  onto  $\mathbb{R}^n$ .

*The sets  $X$  and  $Y$  are then said to be **ambient isotopic**.*

**3.1. Curves for Graphics.** For the curves used in graphics, animation and visualization, it is often assumed that they are non-self-intersecting. Many of the curves used are splines, and the special case of a Bézier curve [33] is defined, below.

**Definition 3.2.** *A Bézier curve is defined by  $\mathbf{b} : [0, 1] \rightarrow \mathbb{R}^n$  for  $n = 1, 2, \dots$  as*

$$\mathbf{b}(u) = \sum_{i=0}^n B_{i,n}(u) \mathbf{P}_i,$$

where

$$B_{i,n}(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i},$$

where the  $B_{i,n}$  are the classical  $n$ -th degree Bernstein polynomials and the  $\mathbf{P}_i$  are known as the **control points**.

Even if a curve is non-self-intersecting, when the geometry changes dynamically during animation or visualization, it is possible to introduce self-intersections. This phenomenon is depicted for a planar Bézier curve in Figure 2, where the transition from left to right is achieved by perturbation of the control points. It is known that a cusp forms prior to self-intersection [33]. However, if the control points are being moved continuously in time, then the appearance of that cusp is only instantaneous, similar to the instantaneous appearance of a self-intersection in Figure 1.

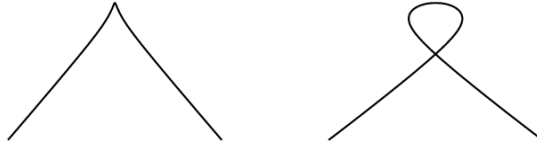


FIGURE 2. Topological Change by Perturbed Control Points

**3.2. Perturbation Limits and Isotopy.** Previous work [6] presented tractable computational limits on the perturbation of control points to preclude introduction of self-intersections. That work is extended here to show that such perturbed curves remain ambient isotopically equivalent. Some of the previous perturbation limits [6] are summarized here for convenience of the reader. For ease of exposition, the summary presented here will only consider the case of a single segment Bézier curve. This captures the essence and the additional subtleties necessary to extend the results here to composite rational Bézier curves follow from the previous work [6]. A topological lemma is presented that shows the existence of an ambient isotopy over this family of non-self-intersecting perturbed curves. This lemma is sufficiently general to apply to any family of perturbed curves that remains non-self-intersecting. Examples will be given over the indicated class of Bézier curves because of the known results for them to be non-self-intersecting [6].

Presenting the previous relevant theory in terminology and notation here relies first upon the definition of the finite sequence  $\mathbf{q}$  defined on the control points of  $\mathbf{b}$  as

$$\mathbf{q} = \{\mathbf{P}_1 - \mathbf{P}_0, \dots, \mathbf{P}_n - \mathbf{P}_{n-1}\}.$$

Let  $\text{conv}(\mathbf{q})$  denote the convex hull of  $\mathbf{q}$  and let  $d^*(\mathbf{q}) = d_E(\mathbf{0}, \mathbf{q})$ , where  $\mathbf{0}$  is the origin and  $d_E$  denotes the Euclidean distance between a point and a polyhedron. The first result used here is the following

**Proposition 3.1.** [6] *A sufficient condition for non-self-intersection of the Bézier curve  $\mathbf{b}$  is that  $d^*(\mathbf{q}) > 0$ .*

While this gives a tractable criterion for determining non-self-intersection of a static Bézier curve, it was also generalized to limit how far the control points could be perturbed without introducing self-intersections. For each  $i = 0, 1, \dots, n$ , let  $\delta\mathbf{P}_i$  represent the perturbation of each control point  $\mathbf{P}_i$ . That is, in analogy to the notation above for  $\mathbf{q}$ , let  $\delta\mathbf{q}$  be given by

$$\delta\mathbf{q} = \{\delta\mathbf{P}_1 - \delta\mathbf{P}_0, \delta\mathbf{P}_n, \dots, \delta\mathbf{P}_{n-1}\}.$$

Let  $\|\delta\mathbf{q}\| = \max\{\|\delta\mathbf{P}_i\| : i = 0, 1, \dots, n-1\}$ , with the Euclidean norm on each  $\delta\mathbf{P}_i$ . The perturbation result used here is the following

**Proposition 3.2.** [6] *If  $d^*(\mathbf{q}) > 0$  and  $d^*(\mathbf{q}) > \|\delta\mathbf{q}\|$ , then the perturbed curve remains non-self-intersecting.*

While this perturbation result only treats a single perturbed curve, it should be clear that if the control points are linearly perturbed to their new positions, then any Bézier curve defined by intermediate positions of those linearly perturbing points is also non-self-intersecting. Linear perturbations of control points are used frequently in graphics, animation and visualization. It is easy to see that each linear perturbation of the control points is a homotopy. The following lemma summarizes that the observed circumstances provide for the existence of an isotopy.

**Lemma 3.1.** *If  $\mathbf{c}$  is a non-self-intersecting curve and  $F$  is a homotopy of  $\mathbf{c}$  such that each homotopic image of  $\mathbf{c}$  is non-self-intersecting, then  $F$  is an ambient isotopy.*

*Proof:* Since  $\mathbf{c}$  is compact, it only remains to show that each homotopic image is also 1 - 1, but this follows from the non-self-intersection. It should also be clear that all the images of  $F$  can be contained within a compact polyhedron so that a standard argument can be made to have an ambient isotopy with compact support [5], merely by extending the original mapping over an arbitrarily small enlargement of the polyhedron.

**3.3. Animating a Curve.** Proposition 3.2 and Lemma 3.1 suffice to exhibit families of Bézier curves that are ambient isotopic within the indicated perturbation limits. If the animation requires perturbation limits that exceed these sufficient conditions, then the perturbation can be partitioned into smaller pieces and each one analyzed separately. So, further attention here will be restricted to a given time interval  $[0, T]$  corresponding to a perturbation less than  $d^*(\mathbf{q})$ .

For any  $t \in [0, T]$ , the partial perturbation at time  $t$  will be ambient isotopic to  $\mathbf{b}$ , where both  $\mathbf{b}$  and its perturbation are Bézier curves. The paradigms for animation and dynamic visualization still require a choice of finitely many times  $\{t_0, t_1, \dots, t_m\}$ , (with  $t_0 = 0$  and each  $t_j \in [0, T]$ ) so displaying these  $m + 1$  images at a sufficiently

high frequency will be perceived as motion. For any such  $t_j$ , the corresponding Bézier curve will be ambient isotopic to the original curve.

However, graphics subsystems typically rely upon piecewise linear (PL) approximations of curves in order to render them. It is well known that PL approximation need not preserve the isotopy class of the original curve [4]. Hence, it remains to determine how ambient isotopic PL approximations can be integrated into this scheme. The first step, of course, is to find an appropriate approximation of  $\mathbf{b}$ .

Recent work by the current authors [25] establishes approximations for  $C^2$  curves  $\mathbf{c} : [0, 1] \rightarrow \mathbb{R}^n$ , where the complexity of the algorithms is expressed in terms of the *bending energy* of  $\mathbf{c}$ , defined as

$$\beta(\mathbf{c}) = \int_0^1 \|\mathbf{c}''(s)\| ds.$$

Previous work [7] has given sufficient conditions for topological complexes of surface patches to be ambient isotopic under perturbations of control points, but this work also does not provide ambient isotopic PL approximations, which is presented for the curves in the next section.

#### 4. APPROXIMATING THE CURVES

Two published algorithms for creating ambient isotopic PL approximation of parametric curves are considered. Both techniques [23, 31] create the same tubular neighborhood of the curve as a containment for the PL approximation. The curves are assumed to have continuous second derivatives, namely, they are  $C^2$ , where, in general,  $C^n$  represents continuous  $n$ -th derivatives. Let  $r$  be the radius of this tubular neighborhood, which is defined in terms of curvature and the critical values of an energy functional. It follows from both these methods that one can construct an ambient isotopic PL approximation of  $C^2$  Bézier curves. However, these methods may create more PL segments than are needed. Recent techniques [25] based upon bending energy have provided refinements so that the number of approximating segments is asymptotically optimal as the bound on the approximation error goes to zero. Since performance of the graphics operations is optimized by limiting the number of segments to be displayed, these refinements are of practical interest.

Let  $\epsilon$  denote the permissible error bound between  $\mathbf{b}$  and its PL approximation, where it is now assumed that  $\mathbf{b}$  is a non-self-intersecting Bézier curve, such that  $d^*(\mathbf{q}) > 0$  and  $d^*(\mathbf{q}) > \|\delta\mathbf{q}\|$ .

The points on  $\mathbf{c}$  defining the PL approximation rely on a partition of  $[0, 1]$  chosen via Taylor's Theorem. For a fixed  $\epsilon > 0$ , a judicious choice of  $h$  is made so that

$$\int_t^{t+h} \|\mathbf{c}''(z)\| dz = \epsilon.$$

Note that asymptotic optimality for this method is obtained as  $\epsilon$  approaches zero via the following result [25].

**Theorem 4.1.** *Let  $\mathbf{c}$  be a  $C^3$  curve and let  $N(\epsilon, \mathbf{c})$  denote the number of approximating linear segments obtained relative to the bending energy. Then*

$$\lim_{\epsilon \rightarrow 0} \sqrt{\epsilon} N(\epsilon, \mathbf{c}) = O(1) \int \sqrt{\|\mathbf{c}''(z)\|} dz.$$

Combining the previous discussion, one easily sees the following result.

**Proposition 4.1.** *If  $\mathbf{b}$  is a  $C^3$  Bézier curve, with*

- $d^*(\mathbf{q}) > 0$  and  $d^*(\mathbf{q}) > \|\delta\mathbf{q}\|$  and
- $\epsilon < r$

*then these perturbations of  $\mathbf{b}$  are ambient isotopic and there exists a PL ambient isotopic approximation of  $\mathbf{b}$  with error bound  $\epsilon$ .*

Now consider the times  $\{t_0, t_1, \dots, t_m\}$  indicated above. The original curve  $\mathbf{b}$  occurs at time  $t_0$ . Let  $\mathbf{b}_j$  denote the perturbed Bézier curve at time  $t_j$ . In order to create an ambient isotopic approximation of each  $\mathbf{b}_j$  by use of the preceding results, one would need to verify the hypothesis that each  $\mathbf{b}_j$  was either  $C^2$  or  $C^3$ , depending upon the approximation technique used. This would permit an arbitrarily close ambient isotopic PL approximation to each  $\mathbf{b}_j$ . However, the example shown in Figure 2 indicates that perturbation of the control points can change the differentiability class. An obvious option would be constraints to preserve the needed differentiability, but this raises questions in differential topology that are beyond the scope of the present paper.

However, Lemma 3.1 provides an alternative in absence of the verification of the needed differentiability. Namely, let  $pl(\mathbf{b}_0)$  denote any PL approximation of  $\mathbf{b}_0$ . Published perturbation limits to prevent self-intersections of PL curves [5] rely on defining a parameter  $\nu$  as the minimum of all distances between disjoint pairs of vertices and segments in the PL curve. Once  $pl(\mathbf{b}_0)$  is known, the determination of its  $\nu$  is a trivial computation. Since, each vertex  $pl(\mathbf{b}_0)$  is a point of  $\mathbf{b}$ , it is easy to determine its perturbed position on  $\mathbf{b}_j$  by the restriction of the perturbation function that mapped  $\mathbf{b}_0$  to  $\mathbf{b}_j$ . Proposition 4.1 can be now rephrased as follows

**Proposition 4.2.** *If  $\mathbf{b}$  is a  $C^3$  Bézier curve, with*

- $d^*(\mathbf{q}) > 0$  and  $d^*(\mathbf{q}) > \|\delta\mathbf{q}\|$ ,
- $\epsilon < r$ , and
- *if  $\mathbf{b}$  has a PL ambient isotopic approximation with  $\nu < \|\delta\mathbf{q}\|$ ,*

*then these perturbations of  $\mathbf{b}$  are ambient isotopic and each has an ambient isotopic PL approximation given as the corresponding perturbation of  $pl(\mathbf{b}_0)$ .*

Note, that these yield a finite family of PL linear curves that are ambient isotopic, but the unsatisfying aspect is that there is no longer any statement about an error bound for the PL approximation of each  $\mathbf{b}_j$ . Indeed, a crude estimate, by the triangle

inequality would only yield an upper bound of  $2\|\delta\mathbf{q}\| + \epsilon$ . However, the restriction of the Bézier curve perturbations to create the PL approximations suggests that, in practice, the errors between  $\mathbf{b}_j$  may be more satisfying, as may be explored by further experimental verification. The hope that finer error bounds would often occur is fueled by the awareness that the time differences between  $t_j$  and  $t_{j+1}$  will typically be quite small and that the user has control over  $\epsilon$  to create a  $pl(\mathbf{b}_0)$  to closely approximate  $\mathbf{b}$ .

The crucial theoretical observation is that attending first to the preservation of topology leads to guarantees for any time chosen within the interval. Only after that, was the interval partitioned to get specific approximations. This avoids the problem of first partitioning the time, only to find that one could have missed a crucial topological change (or even worse, to be oblivious to that pernicious error). The importance of this to molecular simulations [30] is discussed in the following section.

## 5. VISUALIZATION OF MOLECULAR SIMULATIONS

The techniques presented were motivated by applications for dynamic visualization in high performance computing (HPC) environments. A topologically complex geometric model is created to model the resting state of a macro-molecule. For chemical simulations of macro-molecules, the HPC algorithms will produce voluminous numerical data describing how the molecule twists and writhes under local chemical and kinetic changes. These are reflected in changed co-ordinates of the geometric model, namely perturbations of the kind discussed above. To produce a scientifically valid visualization, it is crucial that topological artifacts are not introduced by the visual approximations. Most past attention has focused only upon geometric approximations in each frame appropriate for efficient display. The proposed approach provides bounds for comparison with sufficient conditions to ensure that topological integrity is preserved.

Prototype animations have been implemented on both an open and closed curve. They are available for review over the web [27].

The sufficiency results in some conservatism – there may be perturbations beyond these limits that will still preserve ambient isotopy. However, that is very appropriate to this application. Namely, it now becomes easy to determine when the geometric perturbations approach an indicated limit, merely by a scalar comparison of the distance moved to the perturbation limit. This, alone, may be very valuable to the domain scientists, as this may indicate circumstances where the simulation should be investigated for critical changes. For instance, nearing this limit might signal a time just prior to fracture in a twisting DNA strand [8]. Furthermore, since this comparison is only between scalars, there may be overall performance gains relative to testing each frame for geometric self-intersections, because of the higher performance complexity of those geometric algorithms.

## 6. CONCLUSIONS AND FUTURE WORK

Ambient isotopy is used as a crucial tool for modeling geometric changes at every instant within a time interval. Then ambient isotopic approximations can be created at any selected time within that interval. This reverses the standard animation approach of discretizing the time interval, creating an object at each time instant and then verifying the topology by checking whether new self-intersections may have occurred. Notably, if the time partition had been badly chosen, then critical topological changes could have been missed when the topological checking follows the time partitioning. The attention to topology first, eliminates that problem. Experimental opportunities remain to resolve pragmatic error bounds and performance trade-offs.

## REFERENCES

- [1] P. Agarwal, L. Guibas, A. Nguyen, D. Russel, and L. Zhang. Collision detection for deforming necklaces. *Computational Geometry: Theory and Applications*, 28:137–163, 2004.
- [2] P. K. Agarwal, H. Edelsbrunner, J. Harer, and Y. Wang. Extreme elevation on a 2-manifold. In *SCG '04: Proceedings of the twentieth annual symposium on Computational geometry*, pages 357–365, New York, NY, USA, 2004. ACM Press.
- [3] P. K. Agarwal, H. Edelsbrunner, and Y. Wang. Computing the writhing number of a polygonal knot. In *SODA '02: Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 791–799, Philadelphia, PA, USA, 2002. Society for Industrial and Applied Mathematics.
- [4] N. Amenta, T. J. Peters, and A. C. Russell. Computational topology: ambient isotopic approximation of 2-manifolds. *Theoretical Computer Science*, 305(1-3):3–15, 2003.
- [5] L.-E. Andersson, S. M. Dorney, T. J. Peters, and N. F. Stewart. Polyhedral perturbations that preserve topological form. *Computer Aided Geometric Design*, 12:785–799, 1995.
- [6] L.-E. Andersson, T. J. Peters, and N. F. Stewart. Selfintersection of composite curves and surfaces. *Computer Aided Geometric Design*, 15(5):507–527, 1998.
- [7] L.-E. Andersson, T. J. Peters, and N. F. Stewart. Equivalence of topological form for curvilinear geometric objects. *International Journal of Computational Geometry and Applications*, 10(6):609–622, 2000.
- [8] E. D. T. Atkins and M. A. Taylor. Elongational flow studies on dna in aqueous solution and stress-induced scission of the double helix. *Biopolymers*, 32(8):911 – 923, 1992.
- [9] J. D. Cohen, M. C. Lin, D. Manocha, and M. K. Ponamgi. I-collide: An interactive and exact collision detection system for large-scale environments. In *SI3D*, pages 189–196, 218, 1995.
- [10] L. H. de Figueiredo, J. Stolfi, and L. Velho. Approximating parametric curves with strip trees using affine arithmetic. *Comput. Graph. Forum*, 22(2):171–180, 2003.
- [11] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-Smale complexes for piecewise linear 3-manifolds. In *SCG '03: Proceedings of the nineteenth annual symposium on Computational geometry*, pages 361–370, New York, NY, USA, 2003. ACM Press.
- [12] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse-Smale complexes for piecewise linear 2-manifolds. *Discrete Comput. Geom*, 30:87–107, 2003.
- [13] R. Fleischer, K. Mehlhorn, G. Rote, E. Welzl, and C.-K. Yap. Simultaneous inner and outer approximation of shapes. *Algorithmica*, 8(5&6):365–389, 1992.
- [14] A. Gain and A. Dodgson. Preventing self-intersection under free-form deformation. *IEEE Trans. on Visualization and Computer Graphics*, 7(4):289–298, 2001.

- [15] S. Gottschalk, M. C. Lin, and D. Manocha. Obbtree: A hierarchical structure for rapid interference detection. In *SIGGRAPH*, pages 171–180, 1996.
- [16] N. K. Govindaraju, D. Knott, N. Jain, I. Kabul, R. Tamstorf, R. Gayle, M. C. Lin, and D. Manocha. Interactive collision detection between deformable models using chromatic decomposition. *ACM Trans. Graph.*, 24(3):991–999, 2005.
- [17] L. Guibas. Modeling motion. In J. Goodman and J. O’Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 1117–1134. Chapman and Hall/CRC, 2nd edition, 2004.
- [18] S. Hadap, D. Eberle, P. Volino, M. C. Lin, S. Redon, and C. Ericson. Collision detection and proximity queries. In *SIGGRAPH ’04: ACM SIGGRAPH 2004 Course Notes*, page 15, New York, NY, USA, 2004. ACM Press.
- [19] M. W. Hirsch. *Differential Topology*. Springer-Verlag, New York, 1976.
- [20] H. Kim, L. Guibas, and S. Shin. Efficient collision detection among moving spheres with unknown trajectories. *Algorithmica*, 43(3):195–210, 2005.
- [21] R. Kolodny, L. Guibas, M. Levitt, and P. Koehl. Inverse kinematics in biology: The protein loop closure problem. *Jour. Robotics Research*, 24:151–162, 2005.
- [22] R. Kopperman, M. B. Smyth, D. Spreen, and J. Webster, editors. *Spatial Representation: Discrete vs. Continuous Computational Models*, volume 04351 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany IBFI, Schloss Dagstuhl, Germany, 2005.
- [23] T. Maekawa, N. M. Patrikalakis, T. Sakkalis, and G. Yu. Analysis and applications of pipe surfaces. *Computer Aided Geometric Design*, 15(5):437–458, 1998.
- [24] L. Miller, E. L. F. Moore, T. J. Peters, and A. C. Russell. in press LNCS, 2006.
- [25] L. E. Miller, T. J. Peters, and A. C. Russell. Adaptive curve approximation by bending energy. pre-print, [www.cse.uconn.edu/~tpeters](http://www.cse.uconn.edu/~tpeters).
- [26] E. L. F. Moore. *Computational Topology of Spline Curves for Geometric and Molecular Approximations*. PhD thesis, The University of Connecticut, 2006.
- [27] E. L. F. Moore and T. J. Peters. Related topological animations, images and dissertation. Perturbing . . . , [www.cse.uconn.edu/~tpeters](http://www.cse.uconn.edu/~tpeters).
- [28] E. L. F. Moore and T. J. Peters. Computational topology for geometric design and molecular design. In D. R. Ferguson and T. J. Peters, editors, *Mathematics for Industry: Challenges and Frontiers: A Process View: Practice and Theory*, pages 125–139. Society for Industrial and Applied Mathematics, 2005.
- [29] E. L. F. Moore and T. J. Peters. Floating point geometric algorithms for topologically correct scientific visualization. <http://drops.dagstuhl.de/opus/volltexte/2006/717/>, 2006.
- [30] E. L. F. Moore, T. J. Peters, D. R. Ferguson, and N. F. Stewart. Integrating topology and geometry for macro-molecular simulations. In Kopperman et al. [22].
- [31] E. L. F. Moore, T. J. Peters, and J. A. Roulier. Preserving computational topology by subdivision of quadratic & cubic Bézier curves. *Computing*, 79:317–323, 2007.
- [32] A. Pentland and J. Williams. Good vibrations: model dynamics for graphics and animation. In J. J. Thomas, editor, *SIGGRAPH*, pages 215–222. ACM, 1989.
- [33] L. Piegl and W. Tiller. *The NURBS Book, 2nd Edition*. Springer, New York, NY, 1997.
- [34] D. Russel and L. Guibas. Exploring protein folding conformations using spanners. In *Pacific Symposium on Biocomputing*, pages 40–51, 2005.
- [35] D. W. L. Sumners. Complexity measures for random knots. *Computers & Chemistry*, 14(4):275–279, 1990.
- [36] I. E. Sutherland and G. W. Hodgman. Reentrant polygon clipping. *Commun. ACM*, 17(1):32–42, 1974.