Dynamic Computational Topology for Piecewise Linear Curves *

Hugh P. Cassidy † Thomas J. Peters‡ Kirk E. Jordan§

Abstract

A piecewise linear (PL) approximation often serves as the graphics representation for a parametric curve. Algorithms for preserving correct topology for a single static image are available, but significant challenges remain to ensure correct topology when the PL curve is changing shape during synchronized visualization with an ongoing simulation, such as a molecule writhing over time. A tubular neighborhood of the curve is defined to preserve topology under perturbation, but as the perturbed geometry approaches the boundary of that tubular neighborhood, any required update of the neighborhood should maintain the synchronization. The algorithimic performance of these updates is directly dependent upon the number of approximating edges and the techniques presented here decrease that data volume versus previous methods, as shown by a comprehensive comparative analysis and a representative example.

1 Introduction & Related Work

Molecules undergoing computer simulations are often represented by parametric curves. The simulation code describes how points on the curve move under changes in critical variables such as temperature, pressure and acidity. For graphics display, PL approximations are invoked [10]. The literature on ensuring that a *static* PL approximation retains crucial topological characteristics of the model is relatively well developed for both curves [13] and surfaces [1, 2, 9]. The more subtle challenge of maintaining topological fidelity during perturbations of PL approximations [3] is addressed here, with innovations for efficient update of topological constraints during visualization that is synchronized to an ongoing molecular simulation. These perturbations are constrained within a tubular neighborhood of c, as shown in Figure 1. This tubular neighborhood has a bounding pipe surface [13] of radius r, which depends upon two properties of c,

- the mimimal Euclidean distance between points of c that are distant in arc-length, and
- maximal curvature.

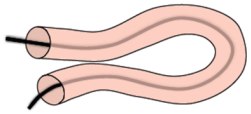

Figure 1: A tubular neighborhood about c.

For polynomial curves, the maximal curvature can be easily computed and will not be discussed further. Our attention is devoted to the first value, known as the *minimum separation distance* for c, abbreviated as MSD_c. A *double normal* is defined to be a line segment which is normal to c at both end points of the line segment. We define MSD_c to be the minimum length of all double normal line segments of c. Even for polynomial curves, the computation of MSD_c entails some subtlety. We show that an approximant of MSD_c, denoted as λ_c can be computed, within user-specified error bounds, from a PL approximation of c. More importantly, we provide a comprehensive comparative analysis and a representative example to justify our efficient updates of λ_c, as the geometry continues to perturb beyond the original constraints imposed by λ_c.

For graphics, once a PL curve has been shown to preserve the topological embedding of the parametric curve, then perturbations of the PL approximation are used to generate subsequent graphical images. By obvious extensions of previous methods [12], the originally calculated λ_c serves as a bound to continue to preserve the desired topological embedding. However, as the limiting value of λ_c is approached, with a perturbed PL curve ℓ, then computation of the updated limit can proceed purely on ℓ.

A concept closely related to these pipe surfaces is the thickness of a knot [5, 6]. Applications to molecular modeling [18] and detailed numeric algorithmic development [4, 7] have appeared, but these algorithms do not address the crucial update efficiencies considered here, even while their definition of *doubly-critical self-distances* is the same as MSD_c.

*The first author was partially supported by NSF Grant CNS 0923158. The second author was partially supported by NSF grants CCF 0429477, CNS 0923158 and CMMI 1053077 as well as by the Joint Study Agreement W1056109 with IBM, an IBM Faculty Award and two IBM Doctoral Fellowships. All statements here are the responsibility of the author, not of the National Science Foundation nor of IBM.

†Department of Computer Science and Engineering, University of Connecticut, Storrs CT, hugh.cassidy@engr.uconn.edu

‡Department of Computer Science and Engineering, University of Connecticut, Storrs CT, tpeters@engr.uconn.edu

§T.J. Watson Research Center, IBM, Cambridge, MA, kjordan@us.ibm.com

2 Preliminaries

We present the curve and topological definitions that are central to this work.

2.1 Class of Parametric Curves

As often occurs, this investigation is first restricted to the class of Bézier curves, as their polynomial representation avoids many cumbersome details, while still supporting theoretical inisights that can easily be generalized to the wider class of non-uniform rational B-spline (NURBS) curves. A degree n Bézier curve with control points, $P = \{p_0, \cdots, p_n\}$ is given by

$$c(t) = \sum_{i=0}^{n} \binom{n}{i} (i-t)^{n-i} p_i, \qquad t \in [0,1],$$

where the PL curve formed by consecutively connecting $p_0, \cdots, p_n$ is called the control polygon of c [17]. A subdivision algorithm operates on P to generate two PL curves, each having $n+1$ vertices, denoted, respectively as P_L and P_R, as shown in Figure 2. The union $P_L \cup P_R$ is also a control polygon for c but lies closer to c than the original control polygon. This process can be repeated to obtain a PL graphical approximation that is within a prescribed distance of the curve c. This is only an initial *static* approximation and the focus here is for methods to ensure that this approximant retains crucial topological characteristics as it changes over time.

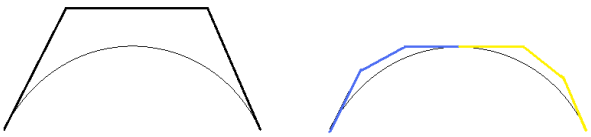

Figure 2: Initial & subdivided control polygons of c.

For ease of exposition[1], we assume that the subdivision parameter is $1/2$, so that the fundamental subdivision operation is to find midpoints of line segments. We remark that the statement, proof and use of Lemma 1 directly depend upon a subdivision parameter of $1/2$.

2.2 Crucial Topological Characteristics

The traditional measure of topological equivalence is *homeomorphism*. A homeomorphism is a mapping, $f : X \longrightarrow Y$, between two subsets X and Y of $\mathbb{R}^n$ such that:

1. f is bijective,
2. f and f^{-1} are continuous.

[1]The reader can modify our results for other parameters.

Homeomorphic equivalence does not capture the embedding of a curve within $\mathbb{R}^3$. In Figure 3 the right image is an unknot and the left is a trefoil. These structures are homeomorphic even though they are embedded differently in $\mathbb{R}^3$.

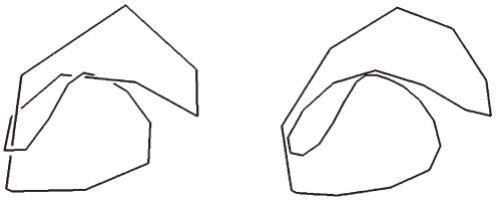

Figure 3: PL knots in $\mathbb{R}^3$

We use the stronger equivalence of *ambient isotopy* to also preserve embedding of c in $\mathbb{R}^3$. The knots in figure 3 are not ambient isotopic. Two subspaces, X and Y, of $\mathbb{R}^n$ are said to be ambient isotopic if there exists a continuous function $H : \mathbb{R}^n \times [0,1] \longrightarrow \mathbb{R}^n$ such that

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

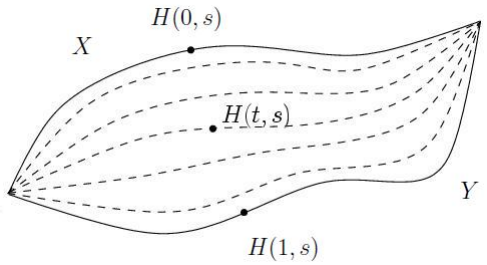

Figure 4: An isotopic deformation of X into Y.

The parameter t can be considered as variable representing time for application to animation and dynamic visualization as illustrated in Figure 4.

3 Background and Notation

In this section we state some previously established results and define notation to be used throughout this paper. Let $c(t)$ be a Bézier curve with control points $P = \{p_0, \cdots, p_n\}$. The PL approximations presented will converge to c in both distance and derivative.

3.1 Approximation of c in Distance

Given the polygon generated by P the *second centered difference* of p_i is given by

$$\Delta_2 p_i = p_{i-1} - 2p_i + p_{i+1}.$$

We define $\Delta_2 p_0 = \Delta_2 p_n = 0$. The *maximal second centered difference* of the polygon generated by P is given by

$$\|\Delta_2 P\|_\infty = \max_{0 \le i \le n} \|\Delta_2 p_i\|_\infty.$$

For a degree n Bézier curve $c(t)$ with control points $P = \{p_0, \cdots, p_n\}$, after m uniform subdivisions the maximal Hausdorff distance between the control polygon and the curve is given by [16]

$$\left(\frac{1}{2}\right)^{2m} \|\Delta_2 P\|_\infty N_\infty(n).$$

Here $N_\infty(n) = \dfrac{\lceil n/2 \rceil \lfloor n/2 \rfloor}{2n}$. Note that this distance is actually attained. So subdividing m_1 times guarantees that the PL structure is within a specified tolerance ϵ where

$$m_1 = \left\lceil -\frac{1}{2} \log_2 \left(\frac{\epsilon}{\|\Delta_2 P\|_\infty N_\infty(n)} \right) \right\rceil \qquad (1)$$

3.2 PL Approximation of c in Derivative

Define the derivative operator Δ on the control points P as follows [11],

$$\begin{aligned} \Delta P &= \{\Delta p_0, \Delta p_1, \cdots, \Delta p_{n-1}\} \\ &= n\{p_1 - p_0, p_2 - p_1, \cdots, p_n - p_{n-1}\}. \end{aligned}$$

The curve generated by the control points ΔP is called the hodograph or derivative curve of c.

Define $\mathcal{L}(P, [0,1])$ to be the uniform parameterization of the control polygon $P = \{p_0, p_1, \cdots, p_n\}$. So

$$\mathcal{L}(P, [0,1]) \left(\frac{j}{n}\right) = p_j$$

and $\mathcal{L}(P, [0,1])$ is linear on the intervals $\left[\frac{j}{n}, \frac{j+1}{n}\right]$. For a Bézier curve $c(t)$ defined on $[0,1]$ with control points $P = \{p_0, \cdots, p_n\}$ the discrete derivative is defined as [15]

$$D[c(t)] = \mathcal{L}(\Delta P, [0,1]).$$

In other words, the discrete derivative is the uniform parameterization of the control polygon of the hodograph.

After applying m subdivisions to $c(t)$ 2^m curves are generated with control points $\left\{p_0^{m,1}, \cdots, p_n^{m,1}, p_0^{m,2}, \cdots, p_n^{m,2}, \cdots, p_0^{m,2^m} \cdots, p_n^{m,2^m}\right\}$. Subdividing m times divides $[0,1]$ into the intervals $\left[\frac{k}{2^m}, \frac{k+1}{2^m}\right]$ for $k = 0, 1, \cdots, 2^m - 1$. Each interval is associated with a unique subcurve. Now define

$$P' = n\left\{\Delta p_0^{k,1}, \Delta p_1^{k,1}, \cdots, \Delta p_{n-1}^{k,1}, \Delta p_0^{k,2}, \cdots, \Delta p_{n-1}^{m,2^m}\right\}.$$

Now write

$$P' = \{p'_0, \cdots, p'_{n2^i}\}$$

The discrete derivative of $c(t)$ after m subdivisions is

$$D_m[c(t)] = \mathcal{L}(P', [0,1]).$$

For a degree n curve subdivided m times we have [8]

$$\|D_m[c(t)] - \frac{d}{dt}c(t)\|_\infty \le \left(\frac{1}{2}\right)^{2m+1} N_\infty(n-1)n\|\Delta_2(\Delta P)\|_\infty.$$

Subdividing m_2 times ensures that we can approximate the derivative within a specified ϵ_d where

$$m_2 = \left\lceil -\frac{1}{2}\left(1 + \log_2 \frac{\epsilon_d}{N_\infty(n-1)n\|\Delta_2(\Delta P)\|_\infty}\right)\right\rceil \qquad (2)$$

3.3 Establishing Double Normals

For a Bézier curve c and distinct $s, t \in [0,1]$ define the quadratic form

$$\langle c(s), c(t)\rangle_{\mathcal{D}} = \frac{[c(s) - c(t)] \cdot c'(s)}{\|c(s) - c(t)\|}.$$

Notice that $c(s)$ and $c(t)$ establish a double normal if and only if

$$\langle c(s), c(t)\rangle_{\mathcal{D}} = \langle c(t), c(s)\rangle_{\mathcal{D}} = 0.$$

The subscript $\mathcal{D}$ here denotes the fact that we are using the continuous derivative.

4 Using PL Structure to Calculate $\mathbf{MSD_c}$

In the previous section we presented a PL approximation to a Bézier curve and its derivatives. Also we defined a quadratic form that we can use to test if given points on the curve form a double normal. The transition to use of the discrete derivative is established through the modified quadratic form

$$\langle c(s), c(t)\rangle_d = \frac{[c(s) - c(t)] \cdot D_m[c(s)]}{\|c(s) - c(t)\|}.$$

The subscript d here indicates that we are using the discrete derivative.

4.1 Testing for Candidate Double Normals

Assume that the user has provided some $\epsilon > 0$ and that we have refined the PL structure with m subdivisions, so it is within $\frac{\epsilon}{2}$ of the curve and the derivative is approximated by the discrete derivative within $\frac{\epsilon}{2}$.

Define $\vec{\gamma} \in \mathbb{R}^3$ so that $D_m[c(s)] = c'(s) + \vec{\gamma}$. Note that $\|\vec{\gamma}\| \le \frac{\epsilon}{2}$. Also, notice that if $c(s)$ and $c(t)$ establish a double normal, i.e. $\langle c(s), c(t)\rangle_{\mathcal{D}} = 0$, then

$$\begin{aligned} \langle c(s), c(t)\rangle_d &= \frac{[c(s) - c(t)] \cdot D_m[c(s)]}{\|c(s) - c(t)\|} \\ &= \frac{[c(s) - c(t)] \cdot [c'(s) + \vec{\gamma}]}{\|c(s) - c(t)\|} \\ &= \frac{[c(s) - c(t)] \cdot \vec{\gamma}}{\|c(s) - c(t)\|} \end{aligned}$$

The Cauchy-Schwarz inequality states that for all vectors x and y of an inner product space it is true that

$$|\langle x, y\rangle| \leq \|x\|\|y\|.$$

So applying the Cauchy-Schwarz inequality yields

$$|\langle c(s), c(t)\rangle_d| \leq \frac{\epsilon}{2}$$

Consider two line segments from the PL representation of c and calculate the minimum distance between the segments. Suppose this minimum distance is realized by the line segment with end points $\mathcal{L}(P,[0,1])(t_0)$ and $\mathcal{L}(P,[0,1])(s_0)$. Then, if

$$|\langle c(s_0), c(t_0)\rangle_d| \leq \frac{\epsilon}{2} \quad \text{and} \quad |\langle c(t_0), c(s_0)\rangle_d| \leq \frac{\epsilon}{2}$$

we consider the line to be a good approximation of a double normal.

4.2 Estimating $\mathbf{MSD_c}$

Denote the exact MSD_c by σ. There exist distinct points $c(s)$ and $c(t)$, such that $d(c(s), c(t)) = \sigma$. We compute λ_c, our estimate for σ as the minimum of the distance between all pairs of disjoint edges of the approximating control polygon. There will exist distinct points p and q from those edges such that $d(p, q) = \lambda_c$.

Lemma 1 *Let ℓ_0 be the length of the longest edge of a given control polygon before any subdivision has occured and ℓ_m be the length of the longest edge after m subdivisions. Then*

$$\ell_m \leq \frac{\ell_0}{2^m}.$$

Proof. This follows from the definition of subdivision with subdivision parameter of 1/2. □

Applying the triangle inequality we see that

$$\sigma = d(c(s), c(t)) \leq d(c(s), p) + d(p, q) + d(q, c(t))$$

Let L_1 be the line segment in the PL approximation that contains the point p, and L_2 be the line segment passing through $c(s)$ with the same length as L_1 so that L_1 and L_2 form opposite sides of a rectangle (see figure 5). Let d_1 denote the diagonal of the rectangle.

We note that $d_1 < d_1^2$, *unless* $d_1 < 1$. In the ensuing analyses, we wish to consider specific numeric bounds on *epsilon*, where we will typically assume that $\epsilon << \ell_m$. To do so, we will make the further simplifying assumption that $\ell_m \geq 1$. Theoretically, the length of each of the finitely many edges could be divided by $\ell_m > 0$, normalizing the measuring scale. Pragmatically, this ensures that the user can conveniently choose values of ϵ and ϵ_d that are small relative to 1.

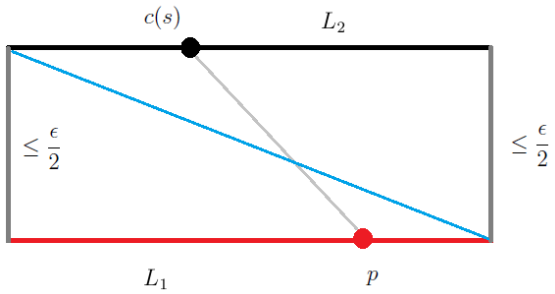

Figure 5: Estimating distance with right triangle

Note that

$$|L_1| = |L_2| \leq \frac{\ell_0}{2^m}$$

and

$$d(c(s), p) \leq d_1.$$

Applying the Pythagorean theorem we have

$$d_1 \leq d_1^2 \leq \left(\frac{\epsilon}{2}\right)^2 + \left(\frac{\ell_0}{2^m}\right)^2 = \frac{\epsilon^2}{4} + \frac{\ell_0^2}{2^{2m}}.$$

So

$$d(c(s), p) \leq \frac{\epsilon^2}{4} + \frac{\ell_0^2}{2^{2m}}.$$

Similarly

$$d(c(t), q) \leq \frac{\epsilon^2}{4} + \frac{\ell_0^2}{2^{2m}}.$$

So we have

$$\begin{aligned} \sigma &\leq 2\left(\frac{\epsilon^2}{4} + \frac{\ell_0^2}{2^{2m}}\right) + \lambda_c \\ &= \frac{\epsilon^2}{2} + \frac{\ell_0^2}{2^{2m-1}} + \lambda_c. \end{aligned}$$

Also,

$$\begin{aligned} \lambda_c &= d(p, q) \\ &\leq d(p, c(s)) + d(c(s), c(t)) + d(c(t), q) \\ &\leq \frac{\epsilon^2}{2} + \frac{\ell_0^2}{2^{2m-1}} + \sigma \end{aligned}$$

So,

$$\sigma \in [\lambda_c - E, \lambda_c + E], \quad \text{where } E = \frac{\epsilon^2}{2} + \frac{\ell_0^2}{2^{2m-1}}.$$

In order to choose a number of subdivisions to minimize E, recall that

$$\left(\frac{1}{2}\right)^{2m} \|\Delta_2 P\|_\infty N_\infty(n) \leq \epsilon,$$

and the fact that ϵ depends on m. Let δ denote our maximum error tolerance, and establish $E \leq \delta$. Let $K = \|\Delta_2 P\|_\infty N_\infty(n)$. Then

$$\left(\frac{1}{2}\right)^{2m} K < \epsilon \quad \text{and} \quad E = 2\left(\frac{\epsilon^2}{4} + \frac{\ell_0^2}{2^{2m}}\right). \quad \text{So}$$

$$E \;<\; 2\left(\frac{\epsilon^2}{4}+\frac{\epsilon\ell_0^2}{K}\right)=\frac{\epsilon^2}{2}+\frac{2\epsilon\ell_0^2}{N_\infty(n)\|\Delta_2 P\|_\infty}$$

So if we choose ϵ so that

$$\frac{\epsilon^2}{2}+\frac{2\epsilon\ell_0^2}{N_\infty(n)\|\Delta_2 P\|_\infty}\le\delta,$$

then clearly $E\le\delta$. This involves solving the following quadratic in ϵ,

$$(1/2)\epsilon^2+\frac{2\ell_0^2\epsilon}{N_\infty(n)\|\Delta_2 P\|_\infty}-\delta\le 0. \tag{3}$$

Choose half the smallest positive root of this quadratic to substitute for ϵ in equation (1).

$\mathbf{MSD_c}$ Estimation Algorithm

Input: δ, c, ϵ_d

0. Calculate m_1, m_2 and let $m=\max\{m_1,m_2\}$.

1. Subdivide m times to get PL approximation of c.

2. Compute $d(l_i,l_j)$, the distances between line segments in the PL structure, where $\mathcal{L}(P,[0,1])(t_0)$ and $\mathcal{L}(P,[0,1])(s_0)$ realize $d(l_i,l_j)$.

3. If $|\langle c(s_0),c(t_0)\rangle_d|\le\epsilon_d$ and $|\langle c(t_0),c(s_0)\rangle_d|\le\epsilon_d$ then keep as double normal.

4. Take minimum from **Step 3.** as λ_c.

Output: λ_c

Figure 6: Algorithm for estimating $\mathrm{MSD_c}$

4.3 Efficient Updates by Data Reduction

The previous approach [14] constructed a PL approximation of c by uniformly partitioning $[0,1]$ as

$$0=s_0<s_1<\cdots<s_{v-1}<s_v=1,$$

where

$$|s_{i+1}-s_i|<\min\left\{\frac{\epsilon}{\sqrt{3}K_0},\frac{\sin\left(\frac{\epsilon}{2}\right)\mu_0}{K_1}\right\}.$$

Here K_0 is the maximum value of $\|c'(t)\|_\infty$, K_1 is the maximum value of $\|c''(t)\|_\infty$ and μ_0 is the minimum value of $\|c'(t)\|_\infty$. The points in the partition are the end points of the curves resulting from subdivision. The number of subdivisions required, $\tilde{m}$, is given by $\tilde{m}=\max\{\tilde{m}_1,\tilde{m}_2\}$ where

$$\tilde{m}_1=\left\lceil-\log_2\left(\frac{\epsilon}{\sqrt{3}K_0}\right)\right\rceil \text{ and}$$

$$\tilde{m}_2=\left\lceil-\log_2\left(\frac{\sin\left(\frac{\epsilon}{2}\right)\mu_0}{K_1}\right)\right\rceil$$

Our algorithm will use fewer subdivisions when $m<\tilde{m}$, given by our comprehensive analysis of 4 cases:
Case 1: $m_1<\tilde{m}_1$. This is true if

$$-\frac{1}{2}\log_2\left(\frac{\epsilon}{\|\Delta_2 P\|_\infty N_\infty(n)}\right)<-\log_2\left(\frac{\epsilon}{\sqrt{3}K_0}\right)+1.$$

In other words if

$$\frac{3K_0^2}{\|\Delta_2 P\|_\infty N_\infty(n)\epsilon}>\frac{1}{4} \tag{4}$$

then $m_1<\tilde{m}_1$. If there is no restriction on ϵ then clearly we can choose ϵ small enough so (4) holds. Otherwise (4) will hold unless K_0 is small, the degree of the curve is very large or $\|\Delta_2 P\|_\infty$ is large (i.e. the control polygon has a narrow spike). So for example if $\epsilon=0.01$ and the curve is cubic

$$\frac{3K_0^2}{\|\Delta_2 P\|_\infty N_\infty(n)\epsilon}=300\left(\frac{M_0^2}{\|\Delta_2 P\|_\infty}\right).$$

So the ratio on the right would need to be less than $\frac{1}{1200}$ for the above inequality to be reversed.
Case 2: $m_2<\tilde{m}_2$ when

$$\frac{\sin\left(\frac{\epsilon}{4}\right)\mu_0\sqrt{n\|\Delta_2(\Delta P)\|_\infty N_\infty(n-1)}}{\sqrt{\epsilon}K_1}<\sqrt{8}$$

This holds unless K_1 is small, μ_0 is large, $\|\Delta_2(\Delta P)\|_\infty$ is large, or the degree is large. So for a cubic with $\epsilon=0.01$ for this inequality to be false $\dfrac{\mu_0}{K_1}>130.6$.
Case 3: $m_1<\tilde{m}_2$ if

$$\frac{\sin\left(\frac{\epsilon}{2}\right)\mu_0\sqrt{\|\Delta_2 P\|_\infty N_\infty(n)}}{K_1\sqrt{\epsilon}}<2.$$

This will not hold for large μ_0, small K_1 or curves of large degree. Note that $\|\Delta_2 P\|_\infty$ is an approximation of K_1.
Case 4: $m_2<\tilde{m}_1$ if

$$\frac{N_\infty(n-1)\|\Delta_2(\Delta P)\|_\infty n\epsilon}{K_0^2}<\frac{8}{3},$$

which holds unless K_0 is small, $\|\Delta_2(\Delta P)\|_\infty$ is large or the degree is large. So outside of the circumstances described above $m<\tilde{m}$. The analysis suggests avoiding 'high degree' curves, as can generally be done in graphics [10]. The other characteristics in the shape of the curve will be assessed in problem specific contexts. Recall, also that the number of approximating segments is exponential in m and $\tilde{m}$, so that even modest differences here can have a significant effect on performance of the topological updating, which depends directly on the number of approximating segments. This is now shown with a representative example.

5 Representative Example

Consider a composite cubic Bézier curve, consisting of five sub-curves, that forms a trefoil knot (figure 7). The control points and calculations are given in the appendix. Choosing $\delta = 0.25$ and $\epsilon_d = 0.01$, gives $\epsilon = 0.01$, so $m = 6$. The maximum error is given by $E = 0.0034415$. This gives a maximum relative error of approximately 0.0078. Note that 6 subdivisions yields 960 line segments. Comparatively the previous approach involved 10,240 line segments [14].

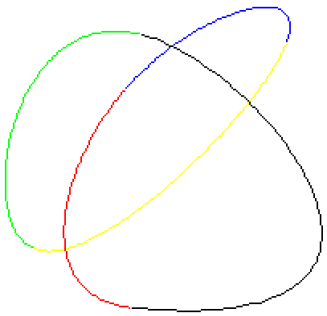

Figure 7: Composite Bézier Curve

6 Conclusion and Future work

We present theory, accompanied by an illustrative example, for efficient updates to constraints for preserving the topological embedding of curves during dynamic visualization of molecular simulations. In high performance computing applications, an accompanying visualization could have millions of frames, so it also becomes important to assess accumulated numerical errors over the total time interval upon realistically challenging data sets. In principle, the dynamic topological results presented here for curves should extend to surfaces, but practical testing on surface data remains a future consideration.

References

[1] N. Amenta and M. Bern. Surface reconstruction by Voronoi filtering. *Discrete and Computational Geometry*, 22:481–504, 1999.

[2] 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.

[3] L.-E. Andersson, T. J. Peters, N. F. Stewart, and S. M. Doney. Polyhedral perturbations that preserve topological form. *Computer Aided Geometric Design*, 12(8):785–799, 1995.

[4] T. Ashton, J. Cantarella, M. Piatek, and E. J. Rawdon. Knot tightening by constrained gradient descent. *Experimental Mathematics*, 20(1):57–90, 2011.

[5] J. Cantarella, G. Kuperberg, R. B. Kusner, and J. M. Sullivan. The second hull of a knotted curve. *American Journal of Mathematics*, 125(6):1335 – 1348, 2003.

[6] J. Cantarella, R. B. Kusner, and J. M. Sullivan. On the minimum ropelength of knots and links. *Inventiones Mathematica*, 150(2):257 – 286, 2002.

[7] J. Cantarella, M. Piatek, and E. Rawdon. Visualizing the tightening of knots. In *VIS '05: Proceedings of the conference on Visualization '05*, pages 575–582, Washington, DC, USA, 2005. IEEE Computer Society.

[8] H. Cassidy and T.J.Peters. Spline operators for subdivision and differentiation, November 2011. pre-print.

[9] K. L. Clarkson. Building triangulations using epsilon-nets. In J. M. Kleinberg, editor, *STOC*, pages 326–335. ACM, 2006.

[10] J. Foley, A. van Dam, S. Feiner, and J. Hughes. *Computer Graphics: Principles and Practice, second edition.* Addison-Wesley Professional, 1990.

[11] J. Gravesen. De Casteljau's algorithm revisited. In *Mathematical methods for curves and surfaces, II (Lillehammer, 1997)*, Innov. Appl. Math., pages 221–228. Vanderbilt Univ. Press, Nashville, TN, 1998.

[12] K. E. Jordan, L. E. Miller, E. L. F. Moore, T. J. Peters, and A. C. Russell. Modeling time and topology for animation and visualization with examples on parametric geometry. *Theoretical Computer Science*, 405:41–49, 2008.

[13] T. Maekawa, N. M. Patrikalakis, T. Sakkalis, and G. Yu. Analysis and applications of pipe surfaces. *Computer Aided Geometric Design*, 15:437–458, 1998.

[14] L. Miller, E. Moore, T. Peters, and A. Russell. Topological neighborhoods for spline curves : Practice and theory. In P. Hertling et al., editors, *Reliable Implementation of Real Number Algorithms: Theory and Practice*, volume 5045 of *LNCS*, pages 149–161. Springer, 2008.

[15] G. Morin and R.Goldman. On the smooth convergence of subdivision and degree elevation for bézier curves. *Computer Aided Geometric Design*, 18:657–666, 2001.

[16] D. Nairn, J. Peters, and D. Lutterkort. Sharp, quantitative bounds on the distance between a polynomial piece and its bézier control polygon. *Computer Aided Geometric Design*, 16:613–631, 1999.

[17] L. Piegl and W. Tiller. *The NURBS Book.* Springer, 2nd edition, 1997.

[18] P. Plunkett, M. Piatek, et al. Total curvature and total torsion of knotted polymers. *Macromolecules*, 40(10):38603867, 2007.

Appendix

In the example we have a composite cubic Bézier curve, consisting of five sub-curves. The control points of each sub-curve are as follows:

$(0,0,0),(-1,1,0),(1.75,3.25,1),(3.21,3.29,2.58)$

$(3.2,3.29,2.58),(4.67,3.33,4.17),(4.83,1.17,6.33)$,
$(3.83,0.67,5.25)$

$(3.83,0.67,5.25),(2.83,0.17,4.17),(0.67,1.33,-.17)$,
$(0.58,2.5,-0.17)$

$(0.58,2.5,-.17),(0.5,3.67,-0.17),(2.5,4.83,4.17)$,
$(3.25,3.04,4.21)$

$(3.25,3.04,4.21),(4,1.25,4.25),(3.5,-3.5,0),(0,0,0)$

From this we can calculate $\ell_0 = 6.393359, \|\Delta_2 P\|_\infty = 9.7532046, \|\Delta_2(\Delta P)\|_\infty = 45.601922$.
Rescaling so $l_0 = 1$ gives $\|\Delta_2 P\|_\infty = 1.5255219$, $\|\Delta_2(\Delta P)\|_\infty = 6.663468$ and $\epsilon_d = \epsilon = .0015641$ So $m = \max\{m_1, m_2\} = \max\{5, 6\} = 6$.