

2004

Computational Geometry Column 45

Joseph O'Rourke

Smith College, jorourke@smith.edu

Follow this and additional works at: https://scholarworks.smith.edu/csc_facpubs

Part of the [Computer Sciences Commons](#), and the [Geometry and Topology Commons](#)

Recommended Citation

O'Rourke, Joseph, "Computational Geometry Column 45" (2004). Computer Science: Faculty Publications, Smith College, Northampton, MA.

https://scholarworks.smith.edu/csc_facpubs/45

This Article has been accepted for inclusion in Computer Science: Faculty Publications by an authorized administrator of Smith ScholarWorks. For more information, please contact scholarworks@smith.edu

Computational Geometry Column 45

Joseph O'Rourke*

Abstract

The algorithm of Edelsbrunner for surface reconstruction by “wrapping” a set of points in \mathbb{R}^3 is described.

Curve reconstruction [O'R00] seeks to find a “best” curve passing through a given finite set of points, usually in \mathbb{R}^2 . Surface reconstruction seeks to find a best surface passing through a set of points in \mathbb{R}^3 . Both problems have numerous applications, usually deriving from the need to reconstruct the curve or surface from a sample. Both problems are highly underconstrained, for there are usually many curves/surfaces through the points. Surface reconstruction in particular is notoriously difficult to control. Although significant advances have been made in recent years [Dey04]—especially in the direction of performance guarantees based on sample density—we turn here to a beautiful and now relatively old “wrapping” algorithm due to Edelsbrunner, which, although implemented in 1996 at Raindrop Geomagic, has been published only recently [Ede03] after issuance of a patent in 2002.

Sample results of the algorithm are illustrated in Figs. 1 and 2.¹ Although both of these examples reconstruct surfaces of genus one, we concentrate on the genus-zero case (a topological sphere) and only mention extensions for higher genus reconstructions.

An attractive aspect of the algorithm is that it reconstructs a unique surface without assumptions on sample density and without adjustment of heuristic parameters. Although the algorithm uses discrete methods, underneath it relies on continuous Morse functions. The discrete scaffolding on

*Dept. of Computer Science, Smith College, Northampton, MA 01063, USA. orourke@cs.smith.edu. Supported by NSF Distinguished Teaching Scholar Grant DUE-0123154.

¹.stl (stereolithography) files for shapes from <http://www.cs.duke.edu/~edels/Tubes/>.

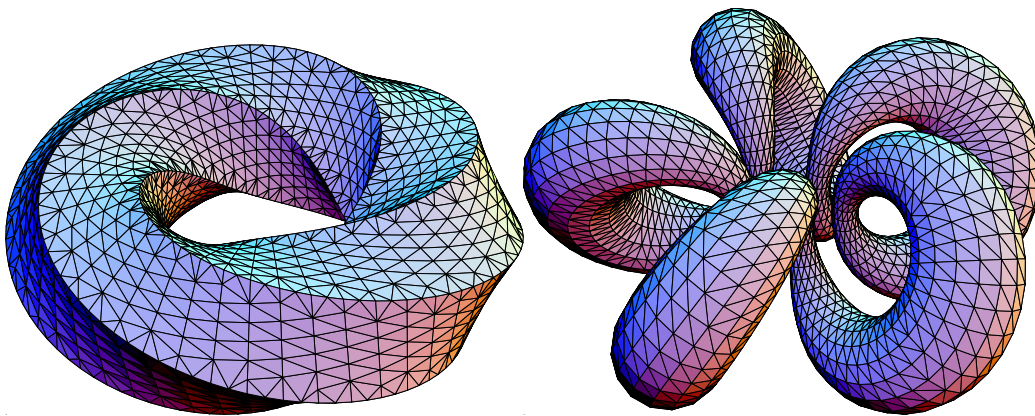


Figure 1: Torus, pentagonal cross-section. Figure 2: Smooth, twisted torus.

which the algorithm depends is the Delaunay complex, which we now informally describe. A *simplex* is a point, segment, triangle, or tetrahedron. A *simplicial complex* \mathcal{K} is a “proper” gluing together of simplices, in that (1) if a simplex σ is in \mathcal{K} , then so are all its faces, and (2) if two simplices σ and σ' are in \mathcal{K} , then either $\sigma \cap \sigma'$ is empty or a face of each. Let S be the finite set of points whose surface is to be reconstructed. The *Delaunay complex* $\text{Del } S$ is the dual of the Voronoi diagram of S . Under a general-position assumption, $\text{Del } S$ contains a simplex that is the convex hull of the sites $T \subset S$ iff there is an empty sphere that passes through the points of T . The outer boundary of $\text{Del } S$ is the convex hull of S . Augmenting $\text{Del } S$ with a dummy “simplex” \emptyset for the space exterior to the hull, covers \mathbb{R}^3 .

The algorithm seeks to find a “wrapping” surface \mathcal{W} , a connected simplicial subcomplex in $\text{Del } S$. It accomplishes this by finding a simplicial subcomplex \mathcal{X} of $\text{Del } S$ whose boundary is \mathcal{W} . The vertices of \mathcal{X} will be precisely the input points S , and the vertices of \mathcal{W} will be a subset of S .

The algorithm uncovers \mathcal{X} in $\text{Del } S$ by “sculpting” away simplices from $\text{Del } S$ one-by-one, starting from \emptyset , until \mathcal{X} remains. The simplices are removed according to an acyclic partial ordering. It is the definition of this ordering that involves continuous mathematics.

A function $g(x)$ assigns to every point $x \in \mathbb{R}^3$ a number dependent on the closest Voronoi vertex. In particular, if x is in a tetrahedron T of $\text{Del } S$ whose empty circumsphere has center z and radius r , then $g(x) = r^2 - \|z - x\|^2$. Thus $g(x)$ is zero at the corners of T and rises to r^2 at z , the closest Voronoi

vertex. Points outside the hull are assigned an effectively infinite value. $g(x)$ is continuous but not smooth enough to qualify as a Morse function, needed for the subsequent development. It will suffice here to claim that g can be smoothed sufficiently to define the vector field ∇g , and from this, by a limiting process, *flow curves* through every point $x \in \mathbb{R}^3$ aiming toward higher values.

These flow curves are in turn used to define an acyclic relation on all the simplices of $\text{Del } S$ and \emptyset . Let τ and σ be two simplices (of any dimension) and v a face shared between them. For example, if τ and σ are both tetrahedra, v could be a triangle, or a segment, or a vertex. Define the *flow relation* “ \rightarrow ” so that $\tau \rightarrow v \rightarrow \sigma$ if there is a flow curve passing from $\text{int } \tau$ to $\text{int } \sigma$.²

A sink of the relation is a simplex that has no flow successor. \emptyset is always a sink (recall $g(x)$ is large outside the hull), with the hull faces of $\text{Del } S$ its immediate predecessors. Sinks are like critical points of the flow, with the simplices that gravitate toward a sink corresponding to a stable manifold in Morse terminology.

A key theorem is that the flow relation on simplices is acyclic, which reflects the increase of $g(x)$ along every flow curve. The algorithm starts with \emptyset and methodically “collapses” its flow predecessors until no more collapses are possible, yielding the complex \mathcal{X} .

Let v be a face of τ ; then τ is called a *coface* of v .³ Assume $\tau \rightarrow v$; for example, τ might be a tetrahedron and v one of its edges, with the flow from τ through v . We give some indication of when the pair (v, τ) is collapsible, without defining it precisely. First, τ must be the highest dimension coface of v , and v should not have any cofaces not part of τ . Thus, v is in a sense “exposed.” Second, the flow curves should pass right through every point of v (as opposed to running along or in v). Collapse of the pair removes all the cofaces of v , thus eating away the parts of τ sharing v .

A second key theorem is that any sequence of collapses from \emptyset leads to the same simplicial complex \mathcal{X} . Collapses also maintain the homotopy type, which, because $\text{Del } S$ is a topological ball, result in \mathcal{X} a ball and \mathcal{W} a topological sphere.

To produce surfaces of higher genus, the contraction is pushed through

² $\text{int } v$ is the interior of v ; for a v a vertex, $\text{int } v = v$.

³One can think of this as a *containing face*, although its origins are more in complementary topological terminology.

holes: the most “significant” sink (in terms of $g(x)$) is deleted (changing the homotopy type), and then the collapses resume as before. This is how the shapes shown in Figs. 1 and 2 were produced. Repeating this process on the sorted sinks results in a series of nested complexes $\mathcal{X} = \mathcal{X}_0, \mathcal{X}_1, \dots, \emptyset$.

Finally, the algorithm works in any dimension, although most applications are in \mathbb{R}^3 .

References

- [Dey04] T. K. Dey. Curve and surface reconstruction. In J. E. Goodman and J. O’Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 30, pages 677–692. CRC Press LLC, second edition, 2004.
- [Ede03] H. Edelsbrunner. Surface reconstruction by wrapping finite point sets in space. In B. Aronov, S. Basu, J. Pach, and M. Sharir, editors, *Discrete and Computational Geometry—The Goodman-Pollack Festschrift*, pages 379–404. Springer-Verlag, 2003.
- [O’R00] J. O’Rourke. Computational geometry column 38. *Internat. J. Comput. Geom. Appl.*, 10(2):221–223, 2000. Also in *SIGACT News*, 31(1):28–30 (2000), Issue 114.