A Simple Provable Algorithm for Curve Reconstruction

Tamal K. Dey   Piyush Kumar*

Abstract

We present an algorithm that provably reconstructs a curve in the framework introduced by Amenta, Bern and Eppstein. The highlights of the algorithm are: (i) it is simple, (ii) it requires a sampling density better than previously known, (iii) it can be adapted for curve reconstruction in higher dimensions straightforwardly.

1 Introduction

We consider the problem of curve reconstruction that takes a set of sample points on a smooth closed curve C, and requires to produce a geometric graph G having exactly those edges that connect sample points adjacent in C. Obviously, given only the samples, it is not always possible to compute G unless some additional conditions are satisfied by the input. Amenta, Bern and Eppstein [1] proposed a framework based on *local feature size* under which they show two graphs, *crust* and *β-skeleton*, coincide with G if the points are sufficiently sampled. Some of the other effective approaches include α-shapes by [6] which is analyzed later by [3], r-regular shapes by [2], $\mathcal{A}$-shapes by [7] and a Delaunay based method by [4]. A survey of these methods appear in [5]. In this paper we show that a modified nearest neighbor graph also coincides with G. The algorithm and its analysis are simple. Nevertheless, it improves the sampling density to 1/3 from 0.252 as required by [1]. More importantly, the algorithm generalizes to higher dimensional curve reconstruction almost straightforwardly. It is not hard to verify that all lemmas and theorem of section 3 hold in any ambient Euclidean space.

We require the following definitions most of which have been introduced in [1]. The *medial axis* M of a smooth curve C in R^d is the closure of all points that have two or more closest points in C. The *local feature size* $f(p)$ at a point $p \in C$ is the least Euclidean distance of p from M. A point set $P \subseteq C$ is an ϵ-sample of C if and only if each point $p \in C$ has a sample within $\epsilon f(p)$ distance. The angle between two edges sharing a common point is the smaller of the two planar angles made by them. We denote the Euclidean distance between two points p, q and the length of an edge e with $\ell(pq)$ and $\ell(e)$ respectively.

*Department of CSE, IIT Kharagpur, Kharagpur 721302, India. e-mail: dey@cse.iitkgp.ernet.in. This research is partially supported by DST, Govt. of India.

2 The algorithm

Algorithm NN-CRUST(input: an ϵ-sample P)

Step 1: Compute the set of edges N that connect nearest neighbors in P.
Step 2: Let a be a point that is incident with only one edge e in N. Compute the shortest edge incident with a among all the edges that make an angle more than $\pi/2$ with e. Let D be the set of all such edges.
Step 3: Output $G = N \cup D$.

Both steps 1 and 2 can be performed on the edges of the Delaunay triangulation T of P since the desired graph G is known to be contained in T [1]. This implies that, in R^2, all steps of NN-CRUST takes time $O(n)$ once T is computed in time $O(n \log n)$, where n is the number of points in P.

3 Proof of correctness

The first lemma is easily deducible from triangular inequality, the second one is proved in [1], and we skip the proof of the third one.

LEMMA 3.1. *$f(q) \leq f(p) + \ell(pq)$ for any two points p, q in C.*

LEMMA 3.2. *If B is a closed ball with $B \cap C$ not a 1-disk, then B contains a medial axis point.*

LEMMA 3.3. *The angle between two adjacent edges in G is more than $\pi/2$ if $\epsilon \leq 1/3$.*

LEMMA 3.4. *$\ell(e) < \frac{2\epsilon}{1-\epsilon} f(p)$ for any edge $e \in G$, where p is an endpoint of e and $\epsilon < 1$.*

Proof: Let q be the point where the perpendicular bisector of $e = ab$ intersects the portion of C over which a and b are adjacent. Grow a ball centered at q until it touches the two endpoints of e. The growing ball always intersects C in a 1-disk since otherwise its radius would be greater than or equal to $f(q)$ (Lemma 3.2) when it had touched the first sample; a case eliminated by the sampling condition at q with $\epsilon < 1$. It follows that the two endpoints of e are the nearest samples to q. This implies $\ell(e) \leq 2\epsilon f(q)$. Substitute $f(q)$ by $\frac{\epsilon}{1-\epsilon} f(p)$ since Lemma 3.1 gives $f(q) \leq f(p) + \ell(pq) \leq f(p) + \epsilon f(q)$.

LEMMA 3.5. *Let $e \notin G$ be any edge between two samples and a be any of its endpoints. Then, either $\ell(e) > f(a)$, or there is an edge $h \in G$ incident with a which makes an angle less than $\pi/2$ with e and $\ell(h) < \ell(e)$.*

Proof: Consider the closed ball B with e as diameter. In case $C_e = B \cap C$ is a 1-disk, there must be an edge $ax \in G$ where x lies in C_e. Otherwise, $e \in G$. It follows that the edge ax sharing an endpoint a with the diameter e must make an angle less than $\pi/2$ with it and $\ell(ax) < \ell(e)$.

In the other case when C_e is not a 1-disk, apply Lemma 3.2 to conclude that B has a medial axis point and hence $\ell(e) > f(a)$.

LEMMA 3.6. *Let a be any sample and b its nearest neighbor. The edge ab is in G if $\epsilon \leq 1/3$.*

Proof: Suppose, on the contrary, $ab \notin G$. Then, we argue that both conditions of Lemma 3.5 are violated reaching a contradiction. Let ax be an edge in G. First consider the case of $\ell(ab) > f(a)$. With $\epsilon \leq 1/3$ we have $\ell(ax) < \frac{2\epsilon}{1-\epsilon} f(a) \leq f(a)$ (Lemma 3.4). This gives $\ell(ax) < \ell(ab)$, an impossibility since b is the nearest neighbor to a. Next, consider the case $\ell(ab) \leq f(a)$. According to Lemma 3.5 there is an edge ax in G so that $\ell(ax) < \ell(ab)$ reaching a contradiction.

THEOREM 3.1. *Given an ϵ-sample for a closed curve with $\epsilon \leq 1/3$, the algorithm* NN-CRUST *outputs an edge e if and only if $e \in G$.*

Proof: Let $e = ab$ be an edge computed by the algorithm. Let ax, ay denote the two edges in G that are incident with a. If e is computed in step 1, it is in G due to Lemma 3.6. Otherwise, it is computed in step 2 which means one of the edges ax and ay, say ax, has already been computed in step 1. The edge e makes an angle more than $\pi/2$ with ax. The edge ay also makes an angle more than $\pi/2$ with ax due to Lemma 3.3. If $e \notin G$, then Lemma 3.5 applies to conclude that $\ell(ay) < \ell(e)$. But, that is impossible since the algorithm chose e to be the shortest edge making angle more than $\pi/2$ with ax.

To show the the other direction consider any edge $e = ab$ in G. If e is a nearest neighbor edge then it is computed in step 1. Otherwise, the other edge in G incident with a, say ax, must be a nearest neighbor edge and has been computed in step 1. The edge e makes an angle more than $\pi/2$ with ax and e is the shortest among all such edges. Otherwise, Lemma 3.5 is violated. This means that e is computed in step 2.

Figure 1: A reconstructed curve in 3D

An example: In Figure 1 we show a reconstruction in $3D$. The 550 points are sampled from the parametric curve $x = \sin t^2$, $y = \cos t^2$, $z = t/3.0$. This is a case of a curve with endpoints. We took care of the endpoints specially in the program.

References

[1] N. Amenta, M. Bern and D. Eppstein. The crust and the β-skeleton: combinatorial curve reconstruction. *Manuscript*, (1997). To appear in *Graphical Models and Image Processing*.

[2] D. Attali. r-regular shape reconstruction from unorganized points. *Proc. 13th Ann. Sympos. Comput. Geom.*, (1997), 248–253.

[3] F. Bernardini and C. L. Bajaj. Sampling and reconstructing manifolds using α-shapes. *Proc. 9th Canadian Conf. Comput. Geom.*, (1997), 193–198.

[4] J. Brandt and V. R. Algazi. Continuous skeleton computation by Voronoi diagram. *Comput. Vision, Graphics, Image Process.* 55 (1992), 329–338.

[5] H. Edelsbrunner. Shape reconstruction with Delaunay complex. *LNCS 1380, LATIN'98: Theoretical Informatics*, (1998), 119–132.

[6] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, vol. 29 (4) (1983), 71–78.

[7] M. Melkemi. $\mathcal{A}$-shapes of a finite point set. *Proc. 13th Ann. Sympos. Comput. Geom.*, (1997), 367–369.