INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Delaunay Triangulation Based Surface Reconstruction: Ideas and Algorithms

Frédéric Cazals — Joachim Giesen

N° 5393

November 2004

Thème SYM

ISSN 0249-6399

Delaunay Triangulation Based Surface Reconstruction: Ideas and Algorithms

Frédéric Cazals * , Joachim Giesen †

Thème SYM — Systèmes symboliques
Projet Geometrica

Rapport de recherche n° 5393 — November 2004 — 42 pages

Abstract: Given a finite sampling $P \subset \mathbb{R}^d$ of an unknown surface S, surface reconstruction is concerned with the calculation of a model of S from P. The model can be represented as a smooth or a triangulated surface, and is expected to match S from a topological and geometric standpoints.

In this survey, we focus on the recent developments of Delaunay based surface reconstruction methods, which were the first methods (and in a sense still the only ones) for which one can precisely state properties of the reconstructed surface. We outline the foundations of these methods from a geometric and algorithmic standpoints. In particular, a careful presentation of the hypothesis used by these algorithms sheds light on the intrinsic difficulties of the surface reconstruction problem —faced by any method, Delaunay based or not.

Key-words: Reverse engineering, Shape approximation, Surface reconstruction, Delaunay, Voronoï

* Project Geometrica, INRIA Sophia
† Departement Informatik, ETH Zürich, Switzerland

Reconstruction de surfaces avec la triangulation de Delaunay: idées forces et algorithmes

Résumé : Etant donné un ensemble de points $P \subset \mathbb{R}^d$ échantillonnés sur une surface (inconnue) S, la reconstruction de surface a pour objet le calcul d'un modèle de S à partir de P. Ce modèle peut être représenté par surface triangulée ou une surface lisse, et doit respecter les propriétés géométriques et topologiques de S.

Ce *survey* a pour objet une présentation des méthodes récentes de reconstruction utilisant la triangulation de Delaunay, méthodes qui ont été les premières pour lesquelles des preuves sur les qualités de la surface reconstruite ont été données. Nous examinons les fondements géométriques mais aussi algorithmiques de ces méthodes. En particulier, un examen minutieux des hypothèses utilisées par ces algorithmes met en évidence les difficultés intrinsèques de la reconstruction de surface, difficultés auxquelles doivent faire face tous les algorithmes —utilisant la triangulation de Delaunay ou pas.

Mots-clés : Ingenierie inverse, Approximation de formes, Reconstruction de surface, Delaunay, Voronoï

1 Introduction

1.1 Surface reconstruction

The surfaces considered in surface reconstruction are two-manifolds that might have boundaries and are embedded in some Euclidean space $\mathbb{R}^d$. In the surface reconstruction problem we are given only a finite sampling $P \subset \mathbb{R}^d$ of an unknown surface S. The task is to compute a model of S from P. This model is referred to as the reconstruction of S from P. It is generally represented as a triangulated surface that can be directly used by downstream computer programs for further processing. The reconstruction should match the original surface in terms of geometric and topological properties. In general surface reconstruction is an ill-posed problem since there are several triangulated surfaces that might fulfill these criteria. Note, that this is in contrast to the curve reconstruction problem where the optimal reconstruction is a polygon that connects the sample points in exactly the same way as they are connected along the original curve. The difficulty of meeting geometric or topological criteria depends on properties of the sampling and on properties of the sampled surface. In particular, sparsity, redundancy, noisiness of the sampling or non-smoothness and boundaries of the surface make surface reconstruction a challenging problem.

Notation. The surface that has to be reconstructed is always denoted by S and a finite sampling of S is denoted by P. The size of P is denoted by n, i.e. $n = |P|$.

1.2 Applications

The surface reconstruction problem naturally arises in computer aided geometric design where it is often referred to as reverse engineering. Typically, the surface of some solid, e.g., a clay mock-up of a new car, has to be turned into a computer model. This modeling stage consists of (i) acquiring data points on the surface of the solid using a scanner (ii) reconstructing the surface from these points. Notice that the previous step is usually decomposed into two stages. First a piece-wise linear surface is reconstructed, and second, piecewise-smooth surface is built upon the mesh.

Surface reconstruction is also ubiquitous in medical applications and natural sciences, e.g., geology. In most of these applications the embedding space of the original surface is $\mathbb{R}^3$. That is why we restrict ourselves in the following to the reconstruction of surfaces embedded in $\mathbb{R}^3$.

1.3 Reconstruction using the Delaunay triangulation

Because reconstruction boils down to establishing neighborhood connections between samples, any geometric construction defining a simplicial complex on these samples is a candidate auxiliary data structure for reconstruction. One such data structure is the Delaunay triangulation of the sample points. The intuition that it might be extremely well suited for reconstruction was first raised in [Boi84] and is illustrated in Figure 1 which features a sampled curve and the Delaunay triangulation of the samples.

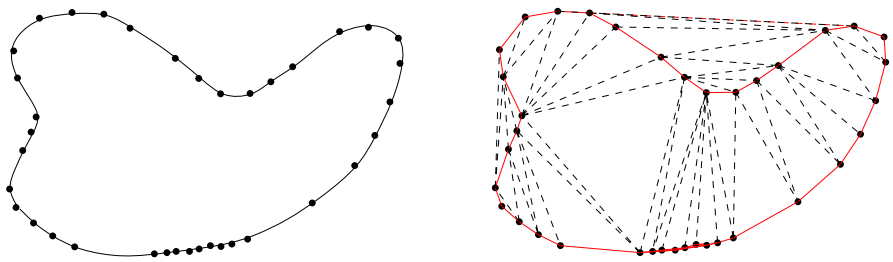

Figure 1: Left: a sampled curve. Right: Delaunay contains a piece-wise linear approximation of the curve. Notice the Delaunay triangulation captures neighbors in *all* directions, no matter how non-uniform the sampling.

The Delaunay triangulation is a cell complex that subdivides the convex hull of the sampling. If the sampling fulfills certain non-degeneracy conditions then all faces in the Delaunay triangulation are simplices and the Delaunay triangulation is unique. The combinatorial and algorithmic complexity of the Delaunay triangulation grow exponentially with the dimension of the embedding space of the original surface. In $\mathbb{R}^3$ the combinatorial as well as the algorithmic complexity of the Delaunay triangulation is $\Theta(n^2)$, where $n = |P|$ is the size of the sampling. However, it has been shown [ABL03] that the Delaunay triangulation of points that are well distributed on a smooth surface has complexity $O(n \log n)$. Robust and efficient methods to compute the Delaunay triangulation in $\mathbb{R}^3$ exist [cga]. Also important for the reconstruction problem is the Voronoi diagram which is dual to the Delaunay triangulation. The Voronoi diagram subdivides the whole space into convex cells where each cell is associated with exactly one sample point.

It seems that the Delaunay triangulation explores the neighborhood of a sample point in all relevant directions in a way that even accommodates non-uniform samplings, see also Figure 1 for a two dimensional example.

There also approaches toward the surface reconstruction problem that are not based on the Delaunay triangulation, e.g., level set methods [HOF01], radial basis function based methods [CBC$^+$01] and moving least squares methods [ABCO$^+$01]. That we do not cover these approaches in this chapter does not mean that they are less suited or worse. On the practical side, many of them are very successfully applied in daily practice. On the theoretical side though, these algorithms often involve non-local constructions making a theoretical analysis difficult. As opposed to these, algorithms elaborating upon Delaunay are more prone to such an analysis, and one of the goals of this survey is to outline the key geometric features involved in these analysis.

1.4 A classification of Delaunay based surface reconstruction methods

Using the Delaunay triangulation still leaves room for quite different approaches to solve the reconstruction problem. But all these approaches, that we sketch below, benefit from the structure of the Delaunay triangulation and the Voronoi diagram, respectively, of the sample points. We should note already here that many of the algorithms combine features of different approaches and as such are not easy to classify. We did the classification by what we consider the dominant idea behind a specific algorithm.

Tangent plane methods. If one considers a smooth surface with a sufficiently dense sampling, the *neighbors* of a point in the point cloud should not deviate too much from the tangent plane of the surface at that point. It turns out that this tangent plane can be well approximated by exploiting the fact that under the condition of sufficiently dense sampling the Voronoi cell of the sample point is elongated in the direction of the surface normal at the sample point. This normal or tangent plane information, respectively, can be used to derive a local triangulation around each point.

Restricted Delaunay based methods. It is possible to define subcomplexes of the Delaunay triangulation by restricting it to some given subset of $\mathbb{R}^3$. Restricted Delaunay based methods compute such a subset from the sampling. This subset should contain the unknown surface S provided the sampling is dense enough. The reconstruction basically is the Delaunay triangulation of P restricted to the computed subset.

Inside / outside labeling. Given a closed surface S one can attempt to classify the tetrahedrons in the Delaunay triangulation as either inside or outside with respect to S. The interface between the inside and outside tetrahedrons should provide a good reconstruction of S. Algorithms that follow the inside / outside labeling paradigm often shell simplices from the outside of the Delaunay triangulation of the sample points in order to discover the surface to be reconstructed. A subclass of the shelling algorithms guide the shelling by topological information like the critical points of some function which can be derived from the sampling.

Empty balls methods. When reconstructing a surface, the simplices reported should be *local* according to some definition. One such definition consists of requiring the existence of a sphere that circumscribes the simplex and does not contain any sample point on its bounded side. The ball bounded by such a sphere is called an empty ball. All Delaunay simplices are local in this sense. This property can be used to filter simplices from the Delaunay triangulation, e.g., by considering the radii of the empty balls.

1.5 Organization of the chapter

The rest of this chapter is subdivided into two sections. Section 2 contains mathematical prerequisites that are necessary to understand the ideas and guarantees behind the algorithms that are detailed in section 3.

2 Pre-requisites

2.1 Delaunay triangulations, Voronoi diagrams and related concepts

General position.

The sampling P is said to be in general position if there are no degeneracies of the following kind: no three points on a common line, no four points on a common circle or hyperplane and no five points on a common sphere. In the following we always assume that the sampling P is in general position. But note that the case that P is not in general position can also be dealt with algorithmically [EM90]. We make the general position assumption only to keep the exposition simple.

Voronoi diagram.

The Voronoi diagram $V(P)$ of P is a cell decomposition of $\mathbb{R}^3$ in convex polyhedrons. Every *Voronoi cell* corresponds to exactly one sample point and contains all points of $\mathbb{R}^3$ that do not have a smaller distance to any other sample point, i.e. the Voronoi cell corresponding to $p \in P$ is given as follows

$$V_p = \{x \in \mathbb{R}^3 : \forall q \in P \quad \|x - p\| \leq \|x - q\|\}.$$

Closed facets shared by two Voronoi cells are called *Voronoi facets*, closed edges shared by three Voronoi cells are called *Voronoi edges* and the points shared by four Voronoi cells are called *Voronoi vertices*. The term *Voronoi object* can denote either a Voronoi cell, facet, edge or vertex. The Voronoi diagram is the collection of all Voronoi objects. See Figure 2 for a two-dimensional example of a Voronoi diagram.

Delaunay triangulation.

The Delaunay triangulation $D(P)$ of P is the dual of the Voronoi diagram, in the following sense. Whenever a collection $V_1, \ldots, V_k$ of Voronoi cells corresponding to point $p_1, \ldots, p_k$ have a non-empty intersection, the simplex whose vertices are $p_1, \ldots, p_k$ belongs to the Delaunay triangulation. It is a simplicial complex that decomposes the convex hull of the points in P. That is, the convex hull of four points in P defines a *Delaunay cell (tetrahedron)* if the common intersection of the corresponding Voronoi cells is not empty. Analogously, the convex hull of three or two points defines a *Delaunay face* or *Delaunay edge*, respectively, if the intersection of their corresponding Voronoi cells is not empty. Every point in P is a *Delaunay vertex*. The term*Delaunay simplex* can denote either a Delaunay cell, face, edge or vertex. See Figure 2 for a two-dimensional example of a Delaunay triangulation.

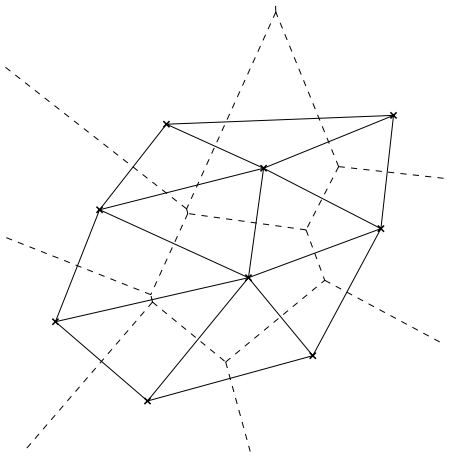

Figure 2: Voronoi and Delaunay diagrams in the plane

Flat tetrahedrons.

In surface reconstruction *flat tetrahedrons* may cause problems for some algorithms. The most notorious flat tetrahedrons are *slivers*. These are Delaunay tetrahedra that have a small volume but do not have a large circumscribing ball and do not have a small edge. Here all comparisons in size are made with respect to the length of the longest edge of the tetrahedron. See Figure 3 for an illustration of a sliver and a cap, and refer to [CDE+99] for a classification of *baldly* shaped tetrahedrons.

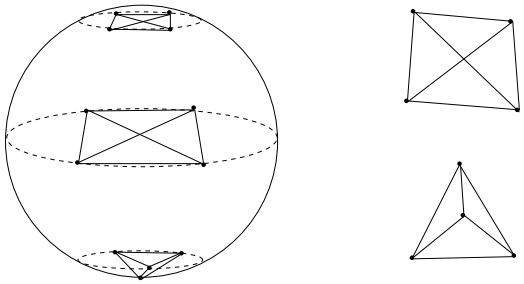

Figure 3: A nearly flat tetrahedron can be located near the equatorial plane or the north pole of its circumscribing sphere. The tetrahedra near the poles have a large circumscribing ball. Only the tetrahedron near the equatorial plane is a sliver (also shown on the top right, the bottom right tetrahedron being a cap.).

Pole.

There are *positive* and *negative poles* associated with a Voronoi cell V_p. If V_p is bounded then the positive pole is the Voronoi vertex in V_p with the largest distance to the sample point p. Let $\mathbf{u}$ be the vector from p to the positive pole. If V_p is unbounded then there is no positive pole. In this case let $\mathbf{u}$ be a vector in the average direction of all unbounded Voronoi edges incident to V_p. The negative pole is the Voronoi vertex v in V_p with the largest distance to p such that the vector $\mathbf{u}$ and the vector from p to v make an angle larger than $\pi/2$.

Empty-ball property.

It follows from the definitions of Voronoi diagrams and Delaunay triangulations that the relative interior of a Voronoi object of dimension k, which is dual to a Delaunay object of dimension $3-k$, consists of the set of points having exactly $3-k+1$ nearest neighbors. Therefore, for any point in such a Voronoi object, there exists a ball empty of sample points containing the vertices of the dual simplex on its boundary. The simplex is said to have the *empty ball property*. See also Figure 4 for a two-dimensional example. For Delaunay tetrahedrons there is only one empty ball whereas there is a continuum of empty balls for Delaunay triangles and edges.

The empty ball property can be used to define sub-complexes of the Delaunay triangulation by imposing additional constraints on the empty balls. Here we discuss two such restrictions that lead to Gabriel simplices and α-shapes, respectively.

Gabriel simplex.

A simplex is called *Gabriel* if its smallest circumscribing ball is empty. Obviously all Gabriel simplices are contained in the Delaunay triangulation. Gabriel simplices also have a dual characterization: a Delaunay simplex is Gabriel iff its dual Voronoi object intersects the affine hull of the simplex.

Well known and heavily used is the Gabriel graph which is the geometric graph that contains all one dimensional Gabriel simplices.

α-shape.

Following the previous discussion, one can attach to each simplex of a Delaunay triangulation an interval of positive real numbers corresponding to the radii of empty balls centered on the Voronoi object dual of the simplex. We refer to this interval as the interval of the simplex. Given a positive real number α, a simplex of dimension $k < 3$ is called *α-exposed* if its interval contains α. The set of α-exposed simplices defines the *α-shape* of the point set. An α-shape usually contains simplices of dimensions between zero and two.

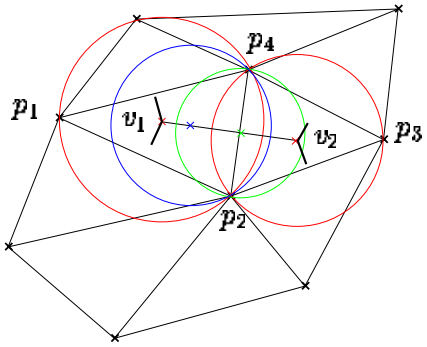

Figure 4: Empty balls centered on Voronoi objects

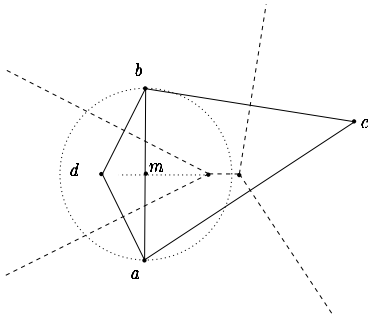

Figure 5: All edges but edge *ab* are Gabriel edges

If we let α grow starting from 0, every Delaunay simplex of dimension smaller than three appears in the evolving α-shape at the lower end of its interval and eventually disappears at the upper end of its interval. Another geometric characterization of the points of appearance and disappearance is as follows: Let balls grow at the sample points with uniform speed. A simplex appears in the α-shape, where α is the evolving radius of the growing balls, when the balls corresponding to the vertices of the simplex intersect for the first time. Note that this intersection happens on the dual Voronoi object of this simplex. It disappears when the common intersection of the balls corresponding to the vertices of the simplex completely contains the dual Voronoi object of the simplex. This growing process is illustrated in Figure 6

Note that α can be interpreted as a spatial scale parameter. If P is a uniform sampling of the surface S then there exist α-values such that the corresponding α-shapes of P provide a reasonable reconstruction of S.

One can also use restricted diagrams that we are going to introduce now to describe an evolving sub-complex of the Delaunay triangulation very similar to α-shapes. Actually the similarity is such that one also refers to these sub-complexes as α-shapes.

Restricted Voronoi diagram and restricted Delaunay triangulation.

Given a subset $X \subset \mathbb{R}^3$. We can restrict the Voronoi diagram of P to to X by replacing every Voronoi object with its intersection with X. The restricted Voronoi diagram is denoted as $V_X(P)$. The Delaunay triangulation $D_X(P)$ of P restricted to X is defined similarly as the Delaunay triangulation of P. The only difference is that instead of taking the common intersection of Voronoi cells now the common intersection of restricted Voronoi cells is taken. That is, whenever a collection $V_1 \cap X, \ldots, V_k \cap X$ of Voronoi cells corresponding to point $p_1, \ldots, p_k$ restricted to X have a non-empty intersection, the simplex whose vertices are $p_1, \ldots, p_k$ belongs to the restricted Delaunay triangulation. The restricted Delaunay triangulation of a plane curve is illustrated in Figure 7.

The Delaunay triangulation restricted to a set of balls with radius α centered at the sample points is called the α-complex. The boundary of the α-complex is the α-shape, and the α value of appearance of Delaunay simplices (of dimension 0 to $d-1$) is the same as for α-shapes. Phrased differently, the differences between the α-complex and the α-shape are twofold: first, once a simplex appears in the α-complex, it stays forever; second, the α-complex also contains Delaunay tetrahedra.

Closed ball property.

The restricted Voronoi diagram $V_S(P)$ of a sampling P of a surface S has the *closed ball property* if the intersection of S with every Voronoi object in $V(P)$ is homeomorphic to a closed ball whose dimension one smaller then that of the Voronoi object. (Notice that the

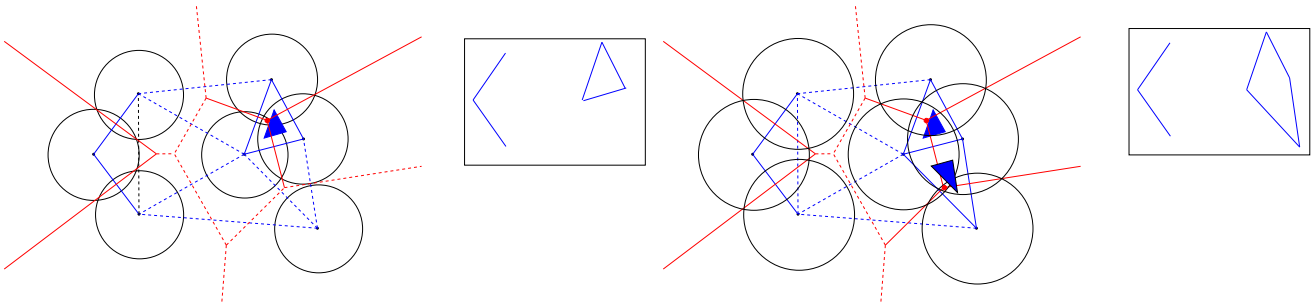

Figure 6: α-complex and α-shapes at two successive times. for each Fig.: the main Fig. and the reduced Fig. feature the α-complex and α-shape

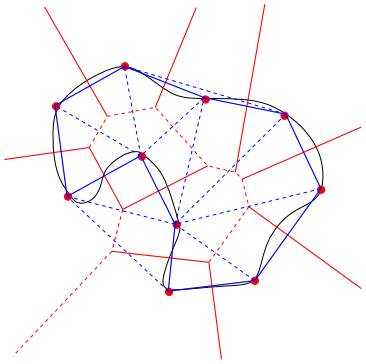

Figure 7: Diagrams restricted to a curve

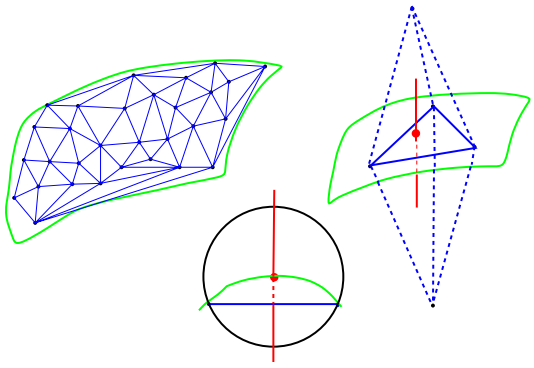

Figure 8: Triangulation restricted to a surface

transverse intersection of a Voronoi cell of dimension k with a manifold of dimension $d-1$ has dimension equal to $k+(d-1)-d = k-1$.) Edelsbrunner and Shah [ES97] were able to relate the topology of the restricted Delaunay triangulation $D_S(P)$ to the topology of S.

Theorem 1 *Let S be a surface and P be a sampling of S such that $V_S(P)$ has the closed all property. Then $D_S(P)$ and S are homeomorphic.*

Power diagram and regular triangulation.

The concepts of Voronoi- and Delaunay diagrams are easily generalized to sets of weighted points. A *weighted point* p in $\mathbb{R}^3$ is a tuple (z,r) where $z \in \mathbb{R}^3$ denotes the point itself and $r \in \mathbb{R}$ its weight. Every weighted point gives rise to a distance function, namely the *power distance function*,

$$\pi_p : \mathbb{R}^3 \to \mathbb{R}, x \mapsto \|x-z\|^2 - r.$$

Let P now be a set of weighted point in $\mathbb{R}^3$. The power diagram of P is a decomposition of $\mathbb{R}^3$ into the *power cells* of the points in P. The power cell of $p \in P$ is given as

$$V_p = \{x \in \mathbb{R}^3 : \forall q \in P, \pi_p(x) \leq \pi_q(x)\}.$$

The points that have the same power distance from two weighted points in P form a hyperplane. Thus V_p is either a convex polyhedron or empty. Closed facets shared by two power cells are called *power facets*, closed edges shared by three power cells are called *power edges* and the points shared by four power cells are called *power vertices*. The term *power object* can denote either a power cell, facet, edge or vertex. The *power diagram* of P is the collection of all power objects.

The dual of the power diagram of P is called the regular triangulation of P. The duality is defined in exactly the same way as for Voronoi diagrams and Delaunay triangulations. That is why regular triangulations are also referred to as weighted Delaunay triangulations.

Natural Neighbors.

Given a Delaunay triangulation, it is natural to define the neighborhood of a vertex as the set of vertices this vertex is connected to. This information is of combinatorial nature and can be made quantitative using the so-called *natural coordinates* which were introduced by Sibson [Sib81].

Given a point $x \in \mathbb{R}^3$ which is not a sample point, define $V^+(P) = V(P \cup \{x\})$, $D^+(P) = D(P \cup \{x\})$, and denote by V_x^+ the Voronoi cell of x in $V^+(P)$. In addition, for any sample point $p \in P$ define $V_{(x,p)} = V_x^+ \cap V_p$ and denote by $w_p(x)$ the volume of $V_{(x,p)}$. The *natural neighbors* of a point x are the the sample points in P that are connected to x in $D^+(P)$. Equivalently, these are the points $p \in P$ for which $V_{(x,p)} \neq \emptyset$. The *natural coordinate*

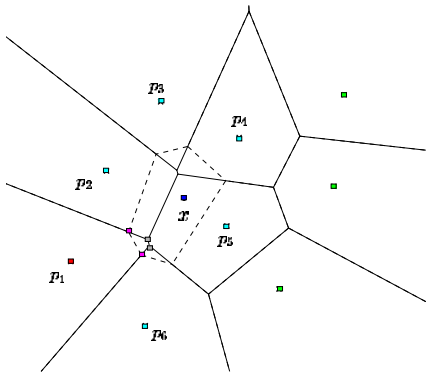

Figure 9: Point x has six natural neighbors

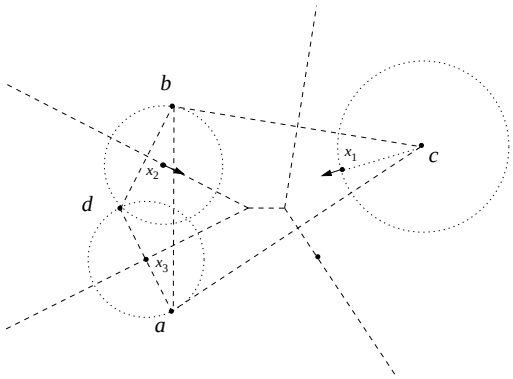

Figure 10: Critical points of the distance function. Points x_1, x_2 are regular, but point x_3 is critical.

associated with a natural neighbor is the quantity

$$\lambda_p(x) = \frac{w_p(x)}{w(x)}, \text{ with } w(x) = \sum_{p \in P} w_p(x). \tag{1}$$

For an illustration of these definitions see Figure 9.

The term *coordinate* is clearly evocative of barycentric coordinates. Recall that in any three-dimensional affine space, a set of four linearly independent points $p_i, i = 1, \ldots, 4$ define a basis of the affine space. Moreover, every point x decomposes uniquely as $x = \sum_{i=1,\ldots,4} \lambda_{p_i}(x) p_i$, with $\lambda_{p_i}(x)$ the barycentric coordinate of x wrt p_i. Natural coordinates provide an elegant extension of barycentric coordinates to the case where one has more than

four linearly points. The following results have been proven in a number of ways [Sib81, Aur88, Bro97, HS02].

Theorem 2 *The natural coordinates satisfy the requirements of a coordinate system, namely,*

(1) for any $p \in P$, $\lambda_p(q) = \delta_{pq}$ where δ_{pq} is the Kronecker symbol and

(2) the point x is the weighted center of mass of its neighbors. That is,

$$x = \sum_{p \in P} \lambda_p(x)\, p, \quad \textit{with} \sum_{p \in P} \lambda_p(x) = 1. \tag{2}$$

Induced distance function.

Voronoi diagrams of a sampling P are closely related to the distance function

$$h : \mathbb{R}^3 \to \mathbb{R}, x \mapsto \min_{p \in P} \|x - p\|$$

induced by the set of sample points. This distance function is smooth everywhere besides at the points in P and on the lower dimensional Voronoi objects, i.e. on the facets, edges and vertices.

At every point x inside a Voronoi region, the gradient of h is the unit vector pointed away from the center of the region. Interestingly, for points x on lower dimensional Voronoi objects, one can in general define a generalized gradient, as depicted on Fig. 10. Let x be a point and denote $C(x)$ the nearest points of x —$C(x)$ consists of the vertices dual of the Voronoi object containing x. If x does not belong to the convex hull of $C(x)$, then the generalized gradient of x points away from the affine hull of $C(x)$. On the other hand, if x belongs to $C(x)$, it is locally impossible to move x so as to increase h, and point x is called a critical point.

It was was observed by Edelsbrunner [Ede04] and later proved by Giesen and John [GJ03] that the critical points of the distance function, i.e., the local extrema and the saddle points, can be characterized in terms of Delaunay simplices and Voronoi objects.

Theorem 3 *The critical points of h are the intersection points of Voronoi objects and their dual Delaunay simplices. The local maxima are Voronoi vertices contained in their dual Delaunay cell. The saddle points are intersection points of Voronoi facets and their dual Delaunay edges and intersection points of Voronoi edges and their dual Delaunay triangles. All sample points are minima.*

The *index* of a critical point is the dimension of the Delaunay simplex involved in its definition. See Figure 2.1 for an example in two dimensions.

Induced flow and stable manifolds.

As in the case of smooth functions there is a unique direction of steepest ascent of h at every non-critical point of h. Assigning to the critical points of h the zero vector and to every other point in $\mathbb{R}^3$ the unique unit vector of steepest ascent defines a *vector field* on $\mathbb{R}^3$. This vector field is non smooth but nevertheless gives rise to a *flow* on $\mathbb{R}^3$, i.e., a mapping

$$\phi : [0, \infty) \times \mathbb{R}^3 \to \mathbb{R}^3,$$

such that at every point $(t, x) \in [0, \infty) \times \mathbb{R}^3$ the right derivative

$$\lim_{t \leftarrow t'} \frac{\phi(t, x) - \phi(t', x)}{t - t'}$$

exists and is equal to the unique unit vector of steepest ascent at x. The flow basically tells how a point would move if it would always follow the steepest ascent of the distance function h. The curve that a point x follows is given by $\phi_x : \mathbb{R} \to \mathbb{R}^3, t \mapsto \phi(t, x)$ and called the *orbit* of x. See Figure 2.1 for some example orbits in two dimensions.

Given a critical point x of h the set of all points whose orbit ends in x, i.e. the set of all points that flow into x, is called the *stable manifold* of x. The collection of all stable manifolds forms a cell complex which is called *flow complex*. See Figure 2.1 for examples of stable manifolds in two dimensions.

2.2 Medial axis and derived concepts

Medial axis.

The medial axis $M(S)$ of a closed subset $S \subset \mathbb{R}^3$ consists of all points in $\mathbb{R}^3 \backslash S$ having two or more nearest points on S. In a way the medial axis generalizes the concept of the Voronoi diagram of a point set. We have seen when discussing the empty ball property that the Voronoi objects of dimension k with $k = 0, \ldots, 2$ consist of all points equidistant from $3 - k + 1$ sample points.

Smooth surfaces S play a special role in reconstruction since for their reconstruction several guarantees can be provided under a certain sampling condition. This sampling condition is based on the medial axis of S that is why we here provide some more details on the structure of the medial axis of a smooth surface S.

Structure of the medial axis of a smooth surface.

The medial axis of a smooth surface S shares another structural property with the Voronoi diagram of a finite point set, namely, it has a stratified structure. For Voronoi diagram this structure means that a Voronoi facet is the common intersection of two Voronoi regions, a Voronoi edge is the common intersection of three Voronoi facets and a Voronoi vertex is the common intersection of four Voronoi edges. To precisely describe the stratified structure of

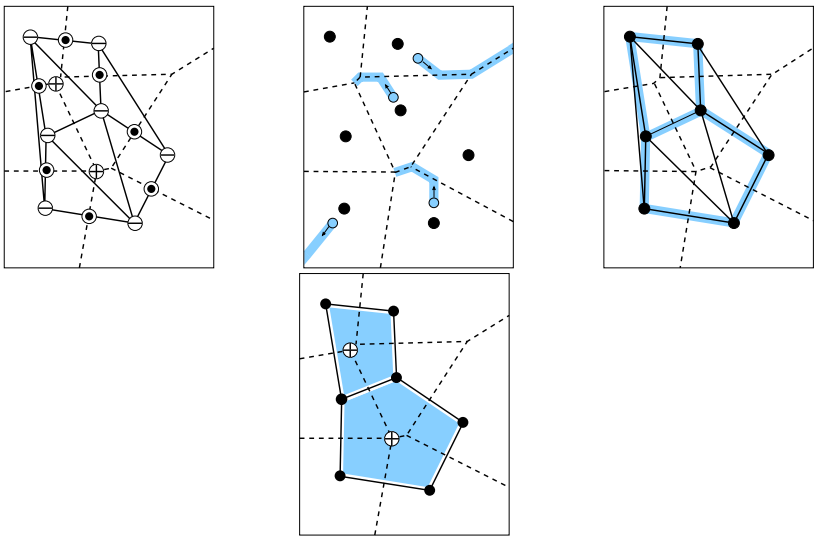

Figure 11: From the left: 1) The local minima $\ominus$, saddle points $\odot$ and local maxima $\oplus$ of the distance function induced by the sample points (local minima). 2) Some orbits of the flow induced by the sample points. 3) The stable manifolds of the saddle points. 4) The stable manifolds of the local maxima.

$M(S)$, one needs the notion of contact between a sphere and the surface. Informally, the contact of a sphere at a point p of S tells how much the sphere and the surface agree at p. More precisely, an A_1 contact means that the tangent plane to the sphere and to S agree at p; an A_2 contact point has the property of an A_1 point with the additional property that the radius of the sphere is the inverse of a principal curvature of S at p; at last, an A_3 contact is like an A_2 contact with the additional property that the curvature involved is an extreme along the corresponding line of curvature. Focusing on the centers of the contact spheres rather than the contact points themselves, and denoting A_1^k a set of $k \geq 1$ simultaneous A_1 contacts between a sphere and the surface, the structure of the $M(S)$ is described by the following theorem [Yom81, GK00] which is illustrated by an example in Figure 12.

Theorem 4 *The medial axis of a smooth surface S in $\mathbb{R}^3$ is a stratified variety containing sheets, curves and points. The sheets correspond to A_1^2 contacts, the curves to A_1^3 and A_3 contacts, and the points to A_1^4 and A_3A_1 contacts. Moreover, one has the following incidences. At an A_1^4 point, six A_2^1 sheets and four A_1^3 curves meet. Along an A_1^3 curve, three A_1^2 sheets meet. A_3 curves bound A_1^2 sheets. At last, the point where an A_1^2 sheet vanishes is an A_3A_1 point.*

Medial axis transform.

A concept closely related to the medial axis of a closed subset $S \subset \mathbb{R}^3$ is the *skeleton* of $\mathbb{R}^3 \backslash S$, which consists of the centers of maximal spheres included in $\mathbb{R}^3 \backslash S$. Here maximal is meant with respect to inclusion among spheres. For a smooth surface S the closure of the medial axis is actually equal to the skeleton of $\mathbb{R}^3 \backslash S$. The *medial axis transform* builds on the close relationship of the skeleton and the medial axis, namely, the medial axis transform is the collection of maximal empty balls centered at the medial axis of S. It can be shown that a smooth surface S can be recovered as the envelope of its medial axis transform.

Tubular neighborhoods.

A natural tool involved in the analysis of several reconstruction algorithms is that of *tubular neighborhood* or *tube* of a surface S. As subsumed by the name, a tube of a surface is a thickening of the surface such that within the volume of the thickening, the projection of a point x to the nearest point $\pi(x)$ on S remains well defined. Following our discussion of the medial axis, a surface can always be thickened provided the thickening avoids the medial axis. Moreover, it is easily checked that the projection onto S proceeds along the normal at the projection point. This property provides a way to *retract* the neighborhood onto the surface.

Feature size.

The *feature size* is a function $f : S \to \mathbb{R}$ on the surface S that assigns to each point in S its least distance to the medial axis of S. An immediate consequence of the triangle inequality is

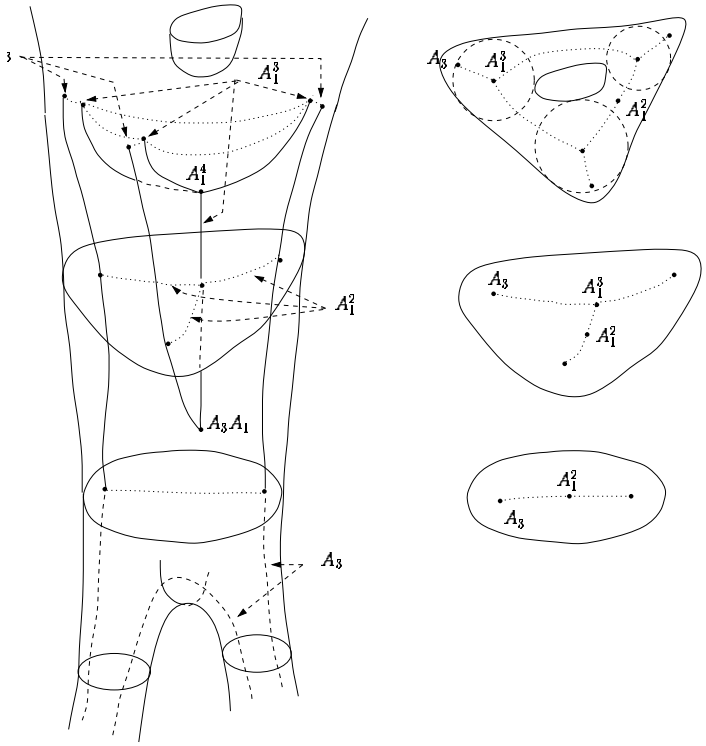

Figure 12: The stratified structure of the medial axis of a smooth surface

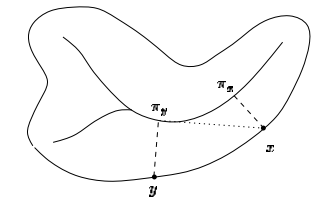

Figure 13: The feature size is 1-Lipschitz

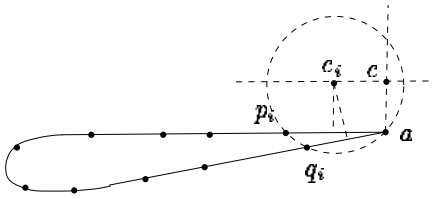

Figure 14: For a non-smooth curve, some Voronoi centers may not converge to the medial axis.

that the feature size of smooth surface is Lipschitz continuous with Lipschitz constant 1, see Figure 13 for an illustration. The feature size can be used to establish another quantitative connection of a surface and its medial axis [BC01b] by using the following theorem.

Theorem 5 *Let B be a ball centered at $x \in \mathbb{R}^3$ with radius r that intersects the surface S. If this intersection is not a topological ball then B contains a point of the medial axis of S.*

¿From this theorem we can conclude that any ball centered at any point $p \in S$ whose radius is smaller then the feature size $f(p)$ at p intersects S in a topological disk.

ε-sampling.

Amenta and Bern [ABE98, AB99] introduced a non-uniform measure of sampling density using the feature size. For $\varepsilon > 0$ a sampling P of a surface S is called an ε-sampling of S if every point x on S has a point of P in distance at most $\varepsilon f(x)$.

We next provide three theorems that involve ε-samples. The first theorem is concerned with the topological equivalence of the restricted Delaunay triangulation $D_S(P)$ and a surface S for an ε-sample P. The second theorem is concerned with the convergence of Voronoi

vertices of the Voronoi diagram of an ε-sample of a smooth surface S towards the medial axis $M(S)$ of S. The last theorem provides a good approximation of the normal of S at some sample point in an ε-sample P.

Amenta and Bern [AB99] proved the following theorem, which provides a topological guarantee for a value of ε less than ~ 0.3. In the context of surface reconstruction, this Theorem should be put in perspective wrt Theorem 1:

Theorem 6 *If P is an ε-sample of S such that ε satisfies*

$$\cos\left(\arcsin\left(\frac{2\varepsilon}{1-\varepsilon}\right)+\frac{\varepsilon}{1-3\varepsilon}\right)>\frac{\varepsilon}{1-\varepsilon}$$

then $V_S(P)$ has the closed ball property.

It can be shown that the Voronoi vertices of a dense sampling of a planar smooth curve lie close to the medial axis of the curve. This result is false in general for non smooth curves, as illustrated in Figure 14. It is also false in general for dense samplings of smooth surfaces. In fact for any point $x \in \mathbb{R}^3 \backslash S$, there exists an arbitrarily dense sampling P of S such that x is a Voronoi vertex of $V(P)$ provided some non-degeneracy holds. To see this grow a ball around x until it touches S. Now grow it a little bit further and put four sample points on the intersection of S with the boundary of the ball. Then x is shared by the Voronoi cells of the four points, i.e., it is a Voronoi vertex if the four points are in general position.

Fortunately, it was observed by Amenta and Bern [AB99] that the poles of the Voronoi diagram of a sampling of a smooth surface converge to the medial axis.

Theorem 7 *Let P be an ε-sample of a smooth surface S the poles of the Voronoi diagram $V(P)$ converge to the medial axis $M(S)$ of S as ε goes to zero.*

Finally, also the following theorem is due to Amenta and Bern [AB99]. It follows from Theorem 7.

Theorem 8 *Let P be an ε-sample of a smooth surface S. For any sample point $p \in P$ let p^+ be the pole of the Voronoi cell V_p. The angle between the normal of S at p and the the vector $p - p^+$ if oriented properly can be bounded by* $\arcsin\left(\frac{2\varepsilon}{1-\varepsilon}\right)$.

2.3 Topological and geometric equivalences

To assess the quality of a reconstruction we need topological and geometric concepts.

Topological concepts.

Homeomorphy. Two surfaces are called *homeomorphic* if there is a *homeomorphism* between them. A *homeomorphism* is a continuous bijection of one surface onto the other, such that the inverse is also continuous. Two homeomorphic surfaces have the same properties

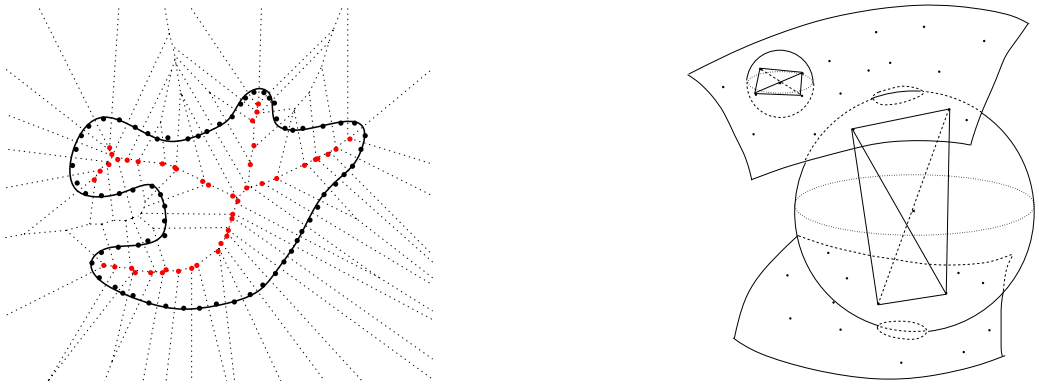

Figure 15: In $2D$, all Voronoi vertices converge to the medial axis. In $3D$, some Voronoi vertices may be far from the medial axis but poles are guaranteed to converge to the medial axis.

regarding open and closed sets, and also neighborhoods. Note that homeomorphy is an equivalence relation. Surfaces that are embedded in $\mathbb{R}^3$ can be completely classified with respect to homeomorphy by their genus, i.e., the number of holes. For example the torus of revolution, i.e., a doughnut, and a "knotted" torus such as are homeomorphic since both have genus 1. This example shows that homeomorphy is a weak concept in the sense that it does not take the ambient space (here $\mathbb{R}^3$) into account. This is done by the concept of *isotopy* which for example accounts for the knottedness of a torus.

Isotopy. Two surfaces are *isotopic* if there exists a one-parameter family of embeddings into $\mathbb{R}^3$ that continuously deform the first surface into the second one. Isotopy is also an equivalence relation. Note the knotted torus can not be deformed continuously into the unknotted one. Any transformation that deforms the knotted torus in the unknotted one has to tear the torus at some point. Thus it can not be continuous.

Homopoty equivalence. If we want to topologically compare the medial axes of two surfaces even the concept of homeomorphy (which can be extended to more complex objects than surfaces) seems too strong since the medial axis of a surface is a more complicated object than the surface itself. For comparing medial axes the concept of *homotopy equivalence* seems to be appropriate. Intuitively, the *homotopy type* of a space encodes its system of internal closed paths, regardless of size, shape and dimension. For example, an annulus has the homotopy type of a circle. Two topological spaces are homotopy equivalent if they have the same homotopy type. Homotopy equivalence is another equivalence relation on topological spaces.

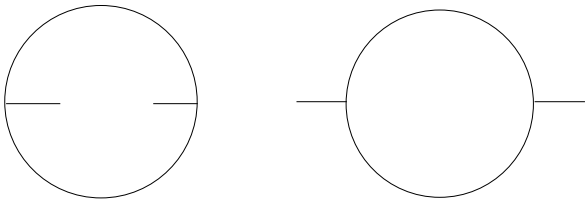

Figure 16: Two homeomorphic topological spaces

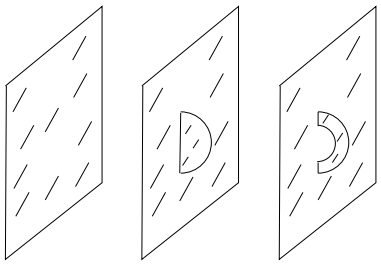

Figure 17: The first two figures have the same homotopy type, but are not homeomorphic. The third one has a different homotopy type.

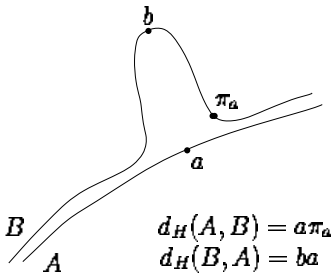

Figure 18: The one-sided Hausdorff distance is not symmetric

Geometric concepts.

Hausdorff distance. The *Hausdorff distance* is a measure for the distance of two subsets of some metric space. We are interested in the case where these subsets are surfaces or medial axes of surfaces in $\mathbb{R}^3$.

Given two closed subsets X, Y of $\mathbb{R}^3$, define the one-sided Hausdorff distance $h(X,Y)$ as $h(X,Y) = \max_{x\in X} \min_{y\in Y} \|x-y\|$. The one-sided Hausdorff distance is not a distance measure since in general it is not symmetric. Symmetrizing h yields the Hausdorff distance defined by $H(X,Y) = \max\{h(X,Y), h(Y,X)\}$. See also Figure 18.

Normals and tangent planes. Given two surfaces, their Hausdorff distance just takes into account their relative positions. In the context of surface reconstruction, we shall also be interested in differential properties of the reconstructed surface with respect to the sampled surface. At the first order, such a measure is provided by the tangent planes (or the normals) to the surfaces, a quantity known to play a key role in the definition of metric properties of surfaces [MT02].

3 Overview of the algorithms

3.1 Tangent plane based methods

We assume the the sampled surface S is smooth, i.e. there exists a well defined tangent plane at each point of the surface. Since we only deal with surfaces of co-dimension one, i.e. surfaces embedded in $\mathbb{R}^3$, approximating the tangent plane at some point of the surface is equivalent to approximating the normal at this point. Thus here tangent plane based methods include normal based methods. The first algorithm based on the tangent planes at the sample points is Boissonnat's [Boi84] algorithm. It probably is the first algorithm at all designed to solve the surface reconstruction problem. Simply put, this algorithm reduces the reconstruction problem to the computation of local reconstructions in the tangent planes at the sample points. These local reconstructions have to be pasted together in the end.

Lower Dimensional Localized Delaunay Triangulation.

Gopi, Krishan and Silva [GKS00] designed an algorithm that is very similar in nature to Boissonnat's early algorithm.

▷ **Bottom-line.** This algorithm has three major steps. First, normal and tangent plane approximation at the sample points. Second, selection of a neighborhood of sample points for each sample point. Third, projection of the neighborhood of a sample point on its tangent plane and computation of the Delaunay neighborhood of the sample point in its projected neighborhood. Sample points $p, q, r \in P$ form a triangle in the reconstruction if they all are mutually contained in their Delaunay neighborhoods.

▷ **Algorithm.** The normal and tangent plane approximation at the sample points is done using the eigenvectors of the covariance matrix of the k nearest neighbors of the sample point p. The covariance matrix is the 3×3 matrix

$$C = \sum_i (q_i - \hat{p})(q_i - \hat{p})^T$$

where the sum is taken over the k nearest neighbors of p in P and $\hat{p}$ is the centroid of the k nearest neighbors of p. The eigenvector corresponding to the smallest eigenvalue of the positive definite, symmetric matrix C is taken as the approximate normal at p. The remaining two eigenvectors span an approximate tangent plane at p. The approximate normals at the sample points are consistently oriented by propagating the orientation at some seed sample point along the edges of the Euclidean minimum spanning tree of P.

¿From the approximated normals at the sample points the directional normal variations and even the principal curvatures at the sample points can be estimated using again the k nearest neighbors of the sample points. The approximated principal curvatures $k_{\min}(p)$ and $k_{\max}(p)$ at a sample point p are used to locally approximate the unknown surface S by a height function

$$h(r, \theta) = \frac{r^2}{2}(k_{\min}(p)\cos^2\theta + k_{\max}(p)\sin^2\theta)$$

parameterized by polar coordinates r and θ over the approximated tangent plane at p. The neighborhood of a sample point p contains all sample points in P at distance at most $2k_{\max}(p)/k_{\min}(p)$ from p whose height value is bounded by some function of $k_{\min}(p)$.

The neighbors of a sample point p are projected onto the approximated tangent plane at p by rotating the vector from p to its neighbor into the tangent plane. In the tangent plane the Delaunay neighbors of p are determined by computing a two dimensional Delaunay triangulation of p and its projected neighbors. The output of the algorithm consists of all triangles with vertices in P whose vertices are mutual Delaunay neighbors.

▷ **Complexity.** The complexity of the algorithm was not theoretically analyzed. But is seems reasonable to assume that the local operations at each sample point can be done in constant time each which would amount to a linear time complexity in total. But there are also the global operations of determining the neighborhoods of the sample points and of

consistently orienting the normals. Though the latter operation is not really needed for the algorithm to work.

▷ **Guarantees.** The triangles output by the algorithm form surface homeomorphic to S provided a curvature based, locally uniform sampling condition holds. This sampling condition also takes care of different parts of S coming close together.

▷ **Extensions.** Some heuristics are given to deal with samplings that do not fulfill the sampling condition. Especially the case of under-sampling is dealt with, though even the extensions do not make sure that the output is topological surface in practice. Of course also oversampling can cause problems since at some points of the algorithm the k nearest neighbors of a sample point are used. This k neighborhood can be spatially biased in the case of oversampling. This bias can invalidate the geometric approximations of normals, tangent planes and curvatures.

Greedy algorithm.

The Greedy algorithm was introduced by Cohen-Steiner and Da in [CSD04]. It incrementally grows a surface from a seed triangle guided by the intuition that the normals vary smoothly over the surface S.

▷ **Bottom-line.** The greedy algorithm incrementally reconstructs an oriented surface $\hat{S}$ by selecting triangles out of the Delaunay triangulation $D(P)$ of P and stitching them to $\hat{S}$. The guideline for the selection is straightforward: the incremental construction should make easy decisions first by stitching triangles which do not yield ambiguities.

▷ **Algorithm.** When extending the surface, a *valid* triangle is a triangle whose stitching does not create a *topological singularity*, and admissible glue operations are of four types *extension, gluing, hole filing, ear filling.*

Let e be a boundary edge of $\hat{S}$. Out of all the valid triangles t incident to e, one of them is chosen as candidate for the surface extension. To define the candidate, denote r_t the radius of the smallest empty ball circumscribing a triangle t. Among all the triangles whose dihedral angle β_t across e is less than some threshold α_s (an angle near π), the candidate is the triangle with least r_t. Since a greedy approach is used, one needs to grade the different candidates. To do so, each triangle is assigned a grade which is $1/r_t$ if β_t is less than a threshold β, and $-\beta_t$ otherwise.
The threshold α_s prevents from considering facets whose stitching would cause a *fold-over* about a large dihedral angle. Notice also the grading strategy favors small triangles provided the dihedral angle is less than the threshold β.

Equipped with these notions, the algorithm consists of the initialization and extension stages. First, the triangle with least circumradius is chosen as a seed, and its edges are pushed into a priority queue Q. Next, the algorithm iterates over Q and processes triangles in order of decreasing confidence. Once a candidate triangle has been popped, a check is performed to see whether a possible extension is possible. This might not be the case

anymore due to potential changes in the environment of the triangle. In any case, the priority queue and the surface are updated.

By construction, the output of the Greedy algorithm is a triangulated and oriented surface, which may not interpolate all the samples since the used thresholds might leave some sample points without incident triangle.

▷ **Complexity.** The algorithm uses the Delaunay triangulation of the samples together with the priority queue. Both data structures determine the complexity of the algorithm.

▷ **Guarantees.** No guarantee can be provided on the quality of the reconstruction due to the difficulty of handling clusters of flat tetrahedrons. As the surface extension is incremental, such clusters can be approached in various manners from different directions, thus making it impossible to close the surface.

▷ **Extensions.** Two heuristics are used to accommodate boundaries as well as sharp features. For boundaries, a candidate triangle is discarded as soon as the radius of its empty ball is significantly larger than that of the triangle it would be stitched to. Sharp edges are detected and removed through the removal of samples which are not part of the output surface.

3.2 Restricted Delaunay based methods

Since all Delaunay based surface reconstruction algorithms filter out a subset of the Delaunay triangulation $D(P)$ of the sampling P it seems natural and very appealing to choose just these simplices from $D(P)$ that are restricted to some subset of $\mathbb{R}^3$ that is a good approximation of the unknown surface S and can be computed efficiently from P. This paradigm is motivated by the fact that if we could directly compute the Delaunay triangulation $D_S(P)$ of P restricted to S we would be done since due to the theorems of Edelsbrunner and Shah (Theorem 1) and Amenta and Bern (Theorem 6), respectively, for sufficiently dense ε-samples $D_S(P)$ is homeomorphic to S.

Crust.

The Crust algorithm was designed by Bern and Amenta [AB99] who also were the first to provide detailed guarantees for the reconstruction provided some ε-sampling condition is fulfilled.

▷ **Bottom-line.** The Crust is based on the Delaunay triangulation $D(P \cup Q)$ of P and the set Q of poles of the Voronoi diagram $V(P)$. Let V be the union of the Voronoi cells of the points in P in the Voronoi diagram $V(P \cup Q)$ of $P \cup Q$. In a nutshell the Crust is the Delaunay triangulation of P restricted to V. The rationale behind this is that $\mathbb{R}^3 \setminus V$ should cover the medial axis $M(S)$ of the surface S. Thus restricting the Delaunay triangulation of P to V should remove all simplices from $D(P)$ that cross the medial axis $M(S)$ of S. On the other hand V should provide a thickened version of S and thus the restricted

Delaunay triangulation $D_V(P)$ should contain all simplices from $D_S(P)$. See Figure 19 for an illustration in two dimensions.

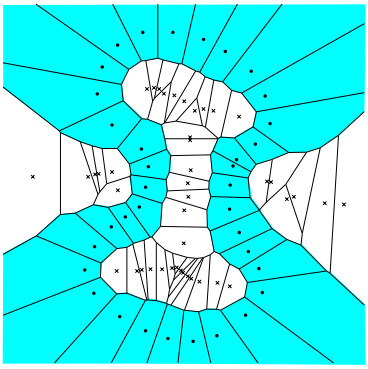

Figure 19: The set V is a thickening of the surface S.

▷ **Algorithm.** The Crust algorithm proceeds as follows: After the set Q of poles is computed from $V(P)$ the Delaunay triangulation $D(P \cup Q)$ of the union of P and Q is constructed. From this Delaunay triangulation all triangles that have all three vertices in P are retained. The retained triangles are called *candidate triangles*. The candidate triangles form not necessarily a surface, but they contain at least one if P is sufficiently dense. One of these surfaces is extracted from the candidate triangles in the final step of the Crust algorithm which walks along the inside or outside of the candidate triangle set and reports all triangles visited. While walking care has to be taken of dangling triangles. These are triangles that have an edge that is incident only to this triangle in the set of candidate triangles. These dangling triangles are recursively removed from the set of candidate triangles before the actual walk starts.

▷ **Guarantees.** It can be shown that for a closed smooth surface S and a sufficiently dense ε-sampling P of S the set of candidate triangles contains all the triangles of the Delaunay triangulation $D_S(P)$ of P restricted to S. This implies that the set of candidate triangles contains at least one surface, namely the restricted Delaunay triangulation $D_S(P)$. Thus the manifold extraction step of the Crust algorithm, i.e., the removal of the dangling triangles from the set of candidate triangles and the actual walking are safe in the sense that the reported triangles actually form a surface.

▷ **Complexity.** Running time and memory consumption of the Crust algorithm are $\Theta(m^2)$, where m is the size of $P \cup Q$. Note, that two Delaunay triangulations have to be computed, one from n and the other from m points. The latter computation determines the asymptotic complexity of the algorithm.

▷ **Extensions.** As described above the Crust algorithm only works for smooth closed surfaces and sufficiently dense ε-samples. Main problem is the final step of the algorithm

that extracts a surfaces from the set of candidate triangles. For practical data sets that do not fulfill the requirements of the algorithm it can happen that this last step removes almost all candidate triangles since dangling triangles are removed recursively. This can be prevented if the removal of the dangling triangles is implemented in a more conservative fashion. This done the Crust algorithm can also cope with surfaces with boundaries and a "certain amount of non-smoothness" in practice.

Cocone.

The Cocone algorithm was designed by Amenta et al. [ACDL00] as a successor and improvement of the Crust algorithm.

▷ **Bottom-line.** The Cocone algorithm builds as the Crust algorithm on the idea of approximating the Delaunay triangulation $D_S(P)$ of P restricted to S by computing a subset $C \subset \mathbb{R}^3$ from P which is a thickened version of S such that the Delaunay triangulation $D_C(P)$ of P restricted to C can computed. The subset C is defined as follows: For every sample point $p \in P$ approximate the normal of S at p using the pole of the Voronoi cell V_p in $V(P)$, see Theorem 8. The co-cone at p is now defined as the intersection of V_p with the complement of a double cone with apex p and fixed opening angle around the approximate normal at p, see Figure 20 for a two dimensional example. The set C is the union of all such co-cones. Note, that C can be computed just from P. Theorem 7 implies that the local thickening of S using co-cones is small compared to the local feature size and thus C is a reasonable approximation of S.

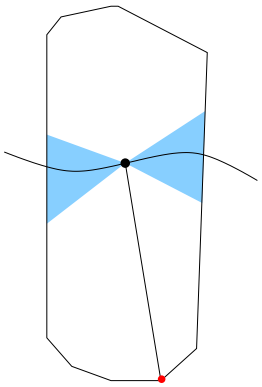

Figure 20: The co-cone of a sample on a curve together with the Voronoi cell of the sample point and its pole.

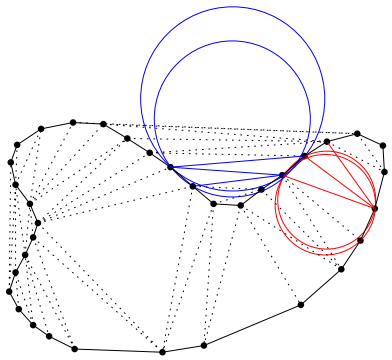

Figure 21: Balls of opposite (the same) color intersect shallowly (deeply)

▷ **Algorithm.** As the Crust algorithm the Cocone algorithm first computes a subset of candidate triangles from the triangles in $D(P)$. A triangle t in $D(P)$ is a candidate triangle if its dual Voronoi edge e intersects any of the co-cones. This intersection test boils down to go through the vertices of t and check if e intersects the co-cone of one of the vertices which is checking the angles the of vectors from a vertex v incident to t to the endpoints of e with the approximate normal at v. As for the Crust algorithm the candidate triangles form not necessarily a surface, but they contain at least one if P is sufficiently dense. Finally, the last step of the Crust algorithm is used to extract one of these surfaces.

▷ **Guarantees.** The same guarantees as for the Crust algorithm hold under the same conditions.

▷ **Complexity.** The running time and memory consumption of the Cocone algorithm is $\Theta(n^2)$ where n is the size of P. This complexity is determined by the computation of $D(P)$. In practice one does not observe the quadratic but a slightly super linear behavior of the running time.

▷ **Extensions.** As described above the Cocone algorithm has the same restrictions as the Crust algorithm which also can be mitigated in the same way. But for the Cocone algorithm there exist a couple of more extensions.

The complexity of the algorithm was reduced by Funke and Ramos to $\Theta(n \log n)$ by avoiding the computation of $D(P)$. They use a data-structure called *well separated pair decomposition* that allows to compute efficiently nearest neighbors of any sample point $p \in P$ in all spatial directions. These neighbors approximate the Voronoi neighbors of p, i.e., the sample points connected to p with an edge in $D(P)$. From these neighbors the normal of S at p can be approximated, and candidate triangles incident to p and two of the approximate Voronoi neighbors can be computed as in the Cocone algorithm.

The output of the Cocone algorithm after making the manifold extraction step robust is a surface with boundary. This surface might contain small unpleasant holes. An extension

called *Tight Cocone* removes these unpleasant holes provided the surface S is closed. The Tight Cocone algorithms falls in the class of inside / outside labeling algorithms, i.e., it removes tetrahedrons from the outside of the Delaunay triangulation $D(P)$. The stopping criterion for the tetrahedron removal is based on the triangles computed by the Cocone algorithm. The latter triangles have to be contained in the reconstruction. The Tight Cocone algorithm was designed by Dey and Goswami [DG03] and got its name from the fact that its output is a watertight surface, i.e., a surface that bounds a solid which might be pinched together at some points.

The Tight Cocone algorithm was even further extended to deal with a noisy sampling P. This extension called *Robust Cocone* was also developed by Dey and Goswami [DG04]. The Robust Cocone algorithm employs a fact that was first used in the Power Crust algorithm of Amenta and Choi [ACK01b, ACK01a], namely, the balls circumscribing adjacent Delaunay tetrahedrons intersect deeply if both tetrahedrons belong to the same component, i.e., either outside or inside. The tetrahedrons only have a shallow intersection if they belong to different components. Dey and Goswami observed that this might not be true for tetrahedrons in the noisy regions around the surface S. But these tetrahedrons have comparatively small circumscribing balls and thus can be detected. In the Robust Cocone algorithm only the sample points on the boundary of the noise layer either facing the outside or the inside are retained. Finally the Tight Cocone algorithm is run on the retained subset of the sample points.

3.3 Inside / outside labeling

The common ancestor of all algorithms that are based on an approximate inside / outside labeling of the Delaunay tetrahedrons with respect to the unknown closed surface S is the algorithm by Boissonnat [Boi84] which probably is the first algorithm at all that addresses the surface reconstruction problem. In his seminal paper Boissonnat uses a sculpturing technique, i.e. removing tetrahedron from the Delaunay triangulation from the outside in order to sculpture a solid whose boundary is the reconstruction. Boissonnat uses a priority queue and weights on the tetrahedrons still present in the shape to decide which tetrahedron to remove next. The tetrahedron removal is controlled by topological constraints, i.e., by prescribing the genus of the surface of the resulting solid.

As depicted on 22, surfaces with boundary do not define an inside and an outside, so that the methods described in this section may not work for such surfaces.

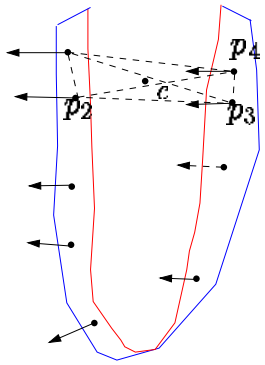

Figure 22: Pole c lies near the medial axis, but cannot be tagged as inside or outside

Power crust.

The Power Crust algorithm was introduced by Amenta et al. in [ACK01b, ACK01a].

▷ **Bottom-line.** To put it briefly, the Power Crust algorithm is an approximate medial axis transform built from empty balls rather than maximal balls, which is asymptotically licit by Theorem 7.

▷ **Algorithm.** Consider a solid T bounded by a closed surface S. By the medial axis transform, the solid T can be expressed as an infinite union of spheres centered on the inner medial axis, i.e., the medial axis of $\mathbb{R}^3 \backslash S$ restricted to T. But if one has a dense point set on S and since the inner poles converge to the inner medial axis, the surface S can certainly be approximated by the boundary of the union of balls centered on the inner poles, their radii being the distance to the corresponding sample points. Similarly, the complement of the solid can be approximated by a union of balls centered on the outer poles. Therefore and since we have two collections of balls (inner and outer), the surface S should also be approximated by the interface between the inner and outer cells of the power diagram defined by the union of inner and outer polar balls. This interface which consists of facets in a power diagram is called the power crust. At last, the dual of the complement of the power crust in the power diagram is called the power shape. The power shape is a subset of the regular triangulation and is expected to approximate the medial axis of $\mathbb{R}^3 \backslash S$.

From an algorithmic standpoint, the critical part consists of tagging the poles as inner or outer. Assuming the samples are enclosed within a bounding box, and starting from the poles of the sample points located on the bounding box, a greedy algorithm is developed. The strategy consists of iterating on the poles using a priority which depends on the angle defined by the two poles of a sample, as well as the intersection angle between polar balls. The algorithm is provably good for dense enough samples, yet may fail in case of under-sampling, at sharp edges or in the presence of noise.

The surface reported is a watertight, i.e., a closed, piecewise linear surface consisting of of power facets, yet possibly pinched at some points. All the sample points are interpolated, yet the result contains additional vertices.

▷ **Complexity.** Three data structures are used: the Delaunay triangulation of the original samples, the power diagram of the poles, and the the priority queue of the tagging algorithm.

▷ **Guarantees.** It is first proved that an inner and an outer ball intersect shallowly —see Fig. 21 for an illustration. This result was already present in [Att98] for the two-dimensional case.

Denoting U_I (U_O) the boundary of the union of inner (outer) balls, the following properties are proved. The one-sided Hausdorff distance between U_I and S is small, and so is the distance between U_O and S, as well as between the power-crust and S. The angle between the normal at a point of U_I or U_O (where it is defined, i.e. on the interior of the spherical caps of U_I and U_O) and the normal at the nearest point on S is also small. This geometric property can be used to show that the projection of U_I (or U_O) to the nearest point on S defines a homeomorphism. Using a similar construction, it is also proved that the power crust and the surface are homeomorphic. At last, the power shape is homotopy equivalent to the complement of the surface. Notice that providing more accurate guarantees on the power shape is significantly more difficult due to the intricate structure of the medial axis, and also due to the fact that the power shape may contain flat tetrahedrons.

▷ **Extensions.** Noise and under-sampling are detected by analyzing the roundness of Voronoi cells. Badly shaped cells and the corresponding poles are discarded. Discarding both poles of a sample which fails the *skinniness* test allows to accommodate sharp edges as the intersection of two facets of the power diagram. At last, large facets of the power crust witnessed by an inner and an outer ball intersecting deeply can be removed, thus leaving a surface with boundary.

Natural Neighbors.

Natural neighbors were first used for reconstruction in [BC00, BC01a]. A reconstruction method based on these algorithms was integrated to CATIA version 5 in 2001.

▷ **Bottom-line.** It is well known by a theorem of Whitney that any smooth surface occurs as the solution set $f^{-1}(0)$ for some smooth function $f : \mathbb{R}^3 \mapsto \mathbb{R}$. The Natural Neighbors reconstruction method is based on the definition of such a function based on two ingredients: an estimate of the tangent plane based on the poles, and the natural coordinates defined with respect to the Voronoi diagram of the samples.

▷ **Algorithm.** For the sake of clarity, first assume that each sample point is given with its normal vector n_i. Denoting $NNs(p)$ the natural neighbors of a point p and $\lambda_i(p)$ the natural coordinate of p with respect to p_i, the method is based upon the following implicit

function $f : \mathbb{R}^3 \mapsto \mathbb{R}$:

$$f(x) = \sum_{p_i \in NNs(p)} \lambda_i \ \langle p_i x, n_i \rangle.$$

The inner product $\langle p_i x, n_i \rangle$ measures the signed distance from x to the tangent plane at p_i, and the function f therefore averages the signed distances to the tangent planes of the natural neighbors of the point x. A direct consequence of the properties of natural coordinates is that the function f interpolates the point cloud, so that the reconstructed surface $\hat{S}$ is naturally defined as $f^{-1}(0)$. It has been conjectured that $\hat{S}$ is a smooth surface, yet it remains to show that 0 is a regular value of f.

Since the natural coordinates have an involved expression, a triangulated approximation of $\hat{S}$ can be obtained as a subset of the Delaunay triangulation of the samples, namely as the restricted Delaunay triangulation $D_{f^{-1}(0)}(P)$. This triangulation is easily computed as follows. Denote by c_1c_2 the dual Voronoi edge of a triangle t. If $f(c_1)f(c_2) < 0$, then triangle t belongs to the restricted Delaunay. Such triangles are also called bipolar in this case.

If the normals are unknown, they can be estimated using the poles. Orienting the normals is also possible using a greedy algorithm similar to the one used in [ACK01b] for the Power-Crust algorithm.

▷ **Complexity.** Apart from the Delaunay triangulation, a priority queue is required to sign the poles and orient the normals if the normals are not provided.

▷ **Guarantees.** It can be shown that the Hausdorff distance between $f^{-1}(0)$ and S tends to zero when the sampling density goes to infinity. There is no guarantee on the topological coherence between the bipolar facets that make up the reconstruction since non manifold edges may be encountered in case of boundaries, noise, or under-sampling. possible.

Topologically guided methods for the inside / outside labeling make use of the distance function induced by the sampling P. In section 2 we have already summarized some properties of this function. In topological guided methods one wants to exploit the critical points of the distance function and their stable manifolds for reconstruction.

Wrap.

The Wrap algorithm was designed by Edelsbrunner [Ede04] already in 1995 and since then marketed by his company.

▷ **Bottom-line.** The Wrap algorithm is based on the concepts of flow and stable manifolds. But instead of building directly on the flow induced by the sample points a *flow relation* is defined on the set of simplices of the Delaunay triangulation $D(P)$ of the sample points P. The critical points are defined exactly as for the distance function induced by P, i.e. as the intersection points of Delaunay- and their dual Voronoi objects. But their stable manifolds are now approximated by subcomplexes of $D(P)$. The reconstruction produced by the Wrap algorithm is the boundary of the union of stable manifolds of a subset of the maxima of the flow relation. As the boundary of a solid it is a surface.

▷ **More details.**

The flow relation $\lhd \subset D \times D$ on the set D of Delaunay simplices is defined as follows: $\tau \lhd \nu \lhd \sigma$ if ν is a face of τ and σ and there exists a point x in the interior of ν such that there is an orbit ϕ_y that is passing from the interior of τ through x to σ. τ is called a *predecessor* and σ is called a *successor* of ν. The relation $\lhd$ is acyclic. A sink is a Delaunay tetrahedron that contains a maximum of the flow, i.e. its dual Voronoi vertex. The set of sinks is augmented by an artificial sink at infinity. The flow relation can be used to define the *ancestor* and *conservative ancestor* sets of a set B of sinks. These sets consist of Delaunay simplices that are that linked to a tetrahedron in B by a chain in the flow relation. The wrapping surface of the point set P is the boundary of the union of the ancestor sets of all finite sinks or equivalently it consists of the complement in the the Delaunay triangulation $D(P)$ of the conservative ancestor set of the sink at infinity. The wrapping surface is unique. It can be computed by collapsing certain simplices. The collapse operation removes simplices that are the unique proper coface of one of their faces. A collapse does not change the homotopy type of the complex since it can be seen as a deformation retraction which always retains the homotopy type. Thus the complex bounded by the wrapping surface is homotopy equivalent to a point, i.e. the wrapping surface cannot be a torus for example. The latter disadvantage is bypassed by allowing a simplex removing operation that changes the homotopy type. The deletion is similar to the original definition of the wrapping surface. Instead of removing from the Delaunay triangulation $D(P)$ only the conservative ancestor set of the sink at infinity, the conservative ancestor sets of a set of sinks is removed. Consequently the wrapping surface is now the boundary of the union of the ancestor sets of the remaining sinks. The latter union need not be homotopy equivalent to a point, i.e. the wrapping surface can be topologically more complicated.

▷ **Guarantees.** No reconstruction guarantees are given besides the fact that the wrapping surface always is the boundary of a solid.

▷ **Complexity.** The running time of the Wrap algorithm is dominated by the time needed to compute the Delaunay triangulation $D(P)$, i.e., it is $\Theta(n^2)$ where n is the size of P.

▷ **Extensions.** No extensions to the Wrap algorithm are known.

Flow complex.

The flow complex is very much related to the Wrap algorithm.

▷ **Bottom-line.** It was observed by Giesen and John [GJ02] that a reconstruction similar to the one obtained by the Wrap algorithm can be derived from the flow complex. The flow complex has a recursive structure, i.e., the stable manifolds of a critical point is bounded by stable manifolds of critical points of lower index. The reconstruction is the boundary of the union of all stable manifolds of the local maxima of the induced distance function. The stable manifold of an index 2 saddle point can be either in the boundary of either one or two stable manifolds of local maxima. As in the Wrap algorithm one can recursively use

stable manifolds of index 2 saddles which are in the boundary of only one stable manifold of a local maximum to push the reconstruction further to interior of the complex. The pushing is guided by considering the difference in value of the height function at the local maximum and the index 2 saddle point.

▷ **Algorithm.** The flow complex is not a subcomplex of the Delaunay triangulation $D(P)$ though $D(P)$ can be used to compute the flow complex. This computation is quite involved and makes use of the recursive structure of the stable manifolds. Here we want to refer the reader to [GJ03] for a detailed description.

▷ **Guarantees.** No reconstruction guarantees are given besides the fact that the wrapping surface always is the boundary of a solid.

▷ **Complexity.** The combinatorial and algorithmic complexities of the flow complex are not known yet. The reconstruction has roughly three times as many triangles as other Delaunay based reconstruction algorithms.

▷ **Extensions.** No extensions are known.

Convection algorithm.

The convection algorithm was designed by Chaine [Cha03].

▷ **Bottom-line.** The Convection algorithm is the geometric implementation of the convection model introduced by Zhao, Osher and Fedkiw []. In this model it is proposed to use a surface $\hat{S}$ as the reconstruction of S from P that minimizes the following energy functional

$$E(S') = \left(\int_{x \in S'} h^p(x) \, \mathrm{d}x \right)^{1/p}, 1 \leq p \leq \infty,$$

where the integral is taken over the closed surface S' and h is the distance function induced by the sampling P. Zhao et al. propose an evolution equation to construct the surface that minimizes the energy functional by deforming a good initial enclosing approximation of the surface. The evolution follows the gradient descent of the energy functional. Every point x of the surface S' evolves towards the interior of the surface along the normal direction $\mathbf{n}$ of S' at x with speed proportional to $-\nabla h(x) \cdot \mathbf{n} + t(x)$, where $t(x)$ is the surface tension of S' at x. To compute an initial approximation of S Zhao et al. change the evolution in the sense the the velocity field at each point $x \in S'$ is replaced by $-\nabla h(x)$. Chaine proves the following theorem for this approximation approach.

Theorem 9 *Given a closed surface S' enclosing the point set P then S' evolves under the convection $-\nabla h(x)$ to a set of closed, piecewise linear pseudo-surfaces. All the facets of these pseudo-surfaces are Delaunay triangles that have the oriented Gabriel property where the triangles are oriented such that their normal points to the inside of the bounded component enclosed by the pseudo-surface.*

A *pseudo surface* can be pinched together along some of its subsets. To define the oriented Gabriel property let t be a Delaunay triangle with oriented normal $\mathbf{n}$. Let s be the half-sphere of the minimum enclosing sphere of t that is contained in half space bounded by the affine hull of t and pointed into by $\mathbf{n}$. The triangle t has the oriented Gabriel property if the half sphere s does not contain any point from P in its interior. Chaine observed that Theorem 9 can be turned into an algorithm based on the Delaunay triangulation $D(P)$ of the sample points P. In this algorithm the evolving pseudo surface is initialized with the boundary of the convex hull of P and all the triangles on this boundary are oriented to point inside the convex hull. The pseudo surface evolves by pushing into Delaunay tetrahedrons. The pushing operations are determined by the vector field $-\nabla h(x)$ and topological constraints.

▷ **Algorithm.** The algorithm basically works as follows: As long as there is a facet f in the evolving, oriented, pseudo surface S' that does not have the oriented Gabriel property do the following. If the facet f with the inverse orientation also belongs to S' then remove f from S'. Otherwise replace f by the three Delaunay facets incident to the Delaunay tetrahedron which is incident to f and on the positive side of f with respect to the orientation of f. Orient the three new facets in S' properly such that their normals point into the interior of the evolving pseudo surface.

▷ **Guarantees.** No reconstruction guarantees are given.

▷ **Complexity.** The running time of the Convection algorithm is dominated by the time needed to compute the Delaunay triangulation $D(P)$, i.e., it is $\Theta(n^2)$ where n is the size of P.

▷ **Extensions.** One modification of the Convection algorithm is to keep an oriented facet f in S' if the same facet with the inverse orientation is also in S'. In doing so the convection algorithm can also reconstruct surfaces with boundaries.

Sometimes the Convection algorithm stops too early, i.e., one would like to push the evolving surface even further. A heuristic to do so is provided.

▷ **Comments.** The Convection algorithm is dual to the Wrap algorithm (and the Flow complex) in the sense that the direction of "flow" is reversed. The Wrap algorithm retains the part of the Delaunay triangulation that does not "flow" to infinity whereas the Convection algorithm lets the convex hull of P "flow" towards the shape.

3.4 Empty balls methods

A triangle reported in a reconstructed surface should be local in some sense. One way to specify locality is to use the empty ball property.

Ball pivoting algorithm

Bernardini et al. designed the ball pivoting algorithm to compute a surface subset of an α-shape of a sampling P in linear time and space [].

▷ **Bottom-line and algorithm.** Like in the definition of α-shapes a triangle pqr with vertices $p, q, r \in P$ forms a triangle in the reconstruction if there is a ball of radius α that contains p, q and r in its boundary and no point from P in its interior. Starting with an α-exposed seed triangle the ball pivoting algorithm pivots around an edge of the seed triangle, i.e., it revolves around an edge while keeping the edge's endpoints on its boundary, until it touches another point from P, forming another triangle. This process continues until all reachable edges have been processed. Then the process continues with a new seed triangle until all points in P have been considered.

▷ **Guarantees.** No guarantees are given.

▷ **Complexity.** Time and space complexity of the ball pivoting algorithm are linear, i.e., it is asymptotically faster than computing the Delaunay triangulation $D(P)$ of P.

▷ **Extensions.** To accommodate non-uniform sampling the pivoting process can be repeated with a larger value for α.

Regular interpolant

The regular interpolant was introduced in [PB01] by Petitjean and Boyer. Their work stresses the importance of Gabriel triangles for surface reconstruction, an observation also raised in [AS00].

▷ **Bottom-line.** The framework of ε-samples might not be the definitive set-up for solving practical problems. To bypass this difficulty, Petitjean and Boyer address the issue of finding an interpolant encoding the properties of the sampling P rather than those of an hypothetical smooth surface S. To see how, we first introduce the relevant notions.

An interpolant O in $\mathbb{R}^3$ is a 2-simplicial complex having P as vertex set. The interpolant is closed if each simplex bounds two distinct connected components of the ambient space. Notice that this definition does not subsume any manifoldness property.

Given a sample point $p \in P$, its *granularity* $g(p)$ is defined as the radius of the largest ball circumscribing a triangle incident to p.

Now, given an interpolant, its associated *discrete medial axis* is the Voronoi diagram from which one removes the Voronoi cells dual to simplices of the interpolant. Notice that the process leaves Voronoi cells of dimensions from two to zero, and in particular all the Voronoi vertices.

The *discrete feature size* or local thickness $t(p)$ at a sample point p is its least distance to the discrete medial axis with the convention $t(p) = 0$ if p is on the boundary of a connected component of $\mathbb{R}^3 \backslash O$ which does not contain any piece of the discrete medial axis.

Equipped with these notions, an interpolant is called *regular* is $g(p) < t(p)$ for all sample points p. Getting back to the point cloud, P is said to be regular if it admits at least one regular interpolant. These notions are depicted in Figure 23.

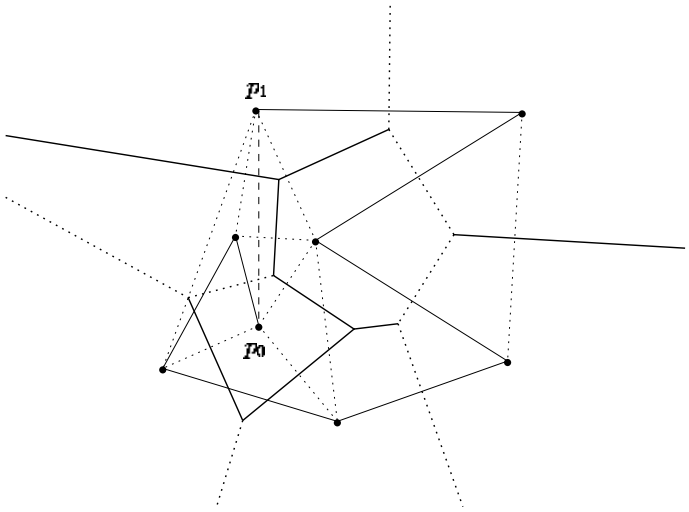

Figure 23: A discrete version of the medial axis. Solid segments: interpolant; dotted segments: belonging to Delaunay but not the interpolant. The medial axis consists of the Voronoi segments dual to Delaunay edges which do not contribute to the interpolant. The Voronoi edges not belonging to the medial axis are dotted too.

Regular interpolants do not exist in general due to the presence of slivers, see Figure 3. When such a tetrahedron is located near its equatorial plane, the granularity is indeed larger than the distance to the discrete medial axis which contains at least the circumcenter of the tetrahedron.

▷ **Algorithm.** For a regular interpolant, the triangles contributing to the interpolant are the Gabriel triangles minimizing the granularity at the vertices. They can be retrieved in an incremental fashion.

▷ **Guarantees.** No guarantees are given.

▷ **Complexity.** Since triangles incident to an edge must be checked for the Gabriel property, the data structure most suited for the reconstruction is the Delaunay triangulation whose computation determines the complexity of the algorithm.

▷ **Extensions.** For non-regular point-sets, triangles are first selected so as to minimize the granularity, and are further decimated if they are not Gabriel. The interpolant built in this way is called a *minimal interpolant*. It is not manifold in general. A manifold extraction step can be applied, which consists of reporting groups of simplices that are simply connected, i.e., contractible to a point.

4 Evaluating surface reconstruction algorithms

Evaluating surface reconstruction algorithms is a difficult task. A prominent feature of some of the algorithms presented in this chapter is that these algorithms come with theoretical guarantees under certain conditions. But if these conditions are not met, their behavior is not specified. Thus it is an interesting question how reconstruction algorithms perform on "real data". In order to assess the performance of different algorithms on real data two surface reconstruction challenges have been organized. One challenge was organized within the *Effective Computational Geometry* project, a project funded by the European Union. The other challenge was organized within a DIMACS Workshop. For both challenges several data sets featuring the following difficulties were selected: undersampling, sharp features, thin parts, boundaries, high genus, noise.

The reader is referred to `www-sop.inria.fr/prisme/manifestations/ECG02/SurfReconsTestbed.html` and `www.cse.ohio-state.edu/dimacs-sr-challenge` where the data sets usded in the challenges are available.

References

[AB99] Nina Amenta and Marshall Bern. Surface reconstruction by Voronoi filtering. *Discrete Comput. Geom.*, 22(4):481–504, 1999.

[ABCO+01] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. Silva. Point set surfaces. In IEEE, editor, *Visualization*, 2001.

[ABE98] Nina Amenta, Marshall Bern, and David Eppstein. The crust and the β-skeleton: Combinatorial curve reconstruction. *Graphical Models and Image Processing*, 60:125–135, 1998.

[ABL03] D. Attali, J.-D. Boissonnat, and A. Lieutier. Complexity of the delaunay triangulation of points on surfaces the smooth case. In *ACM SoCG*, San Diego, 2003.

[ACDL00] N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. In *Proc. 16th Annu. ACM Sympos. Comput. Geom.*, pages 213–222, 2000.

[ACK01a] N. Amenta, S. Choi, and R. Kolluri. The power crust. In *ACM Solid Modeling*, 2001.

[ACK01b] N. Amenta, S. Choi, and R. K. Kolluri. The power crust, unions of balls, and the medial axis transform. *Comput. Geom. Theory Appl.*, 19:127–153, 2001.

[AS00] M. Attene and M. Spagnuolo. Automatic surface reconstruction from point sets in space. In *Eurographics*, 2000.

[Att98] D. Attali. r-regular shape reconstruction from unorganized points. *Comput. Geom. Theory Appl.*, 10:239–247, 1998.

[Aur88] F. Aurenhammer. Linear combinations from power domains. *Geom. Dedicata*, 28:45–52, 1988.

[BC00] Jean-Daniel Boissonnat and Frédéric Cazals. Smooth surface reconstruction via natural neighbour interpolation of distance functions. In *Proc. 16th Annu. ACM Sympos. Comput. Geom.*, pages 223–232, 2000.

[BC01a] Jean-Daniel Boissonnat and Frédéric Cazals. Coarse-to-fine surface simplification with geometric guarantees. In A. Chalmers and T.-M. Rhyne, editors, *Eurographics'01*, Manchester, 2001. Blackwell.

[BC01b] Jean-Daniel Boissonnat and Frédéric Cazals. Natural coordinates of points on a surface. *Comput. Geom. Theory Appl.*, 19:155–173, 2001.

[Boi84] Jean-Daniel Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Trans. Graph.*, 3(4):266–286, 1984.

[Bro97] J. L. Brown. Systems of coordinates associated with points scattered in the plane. *Comput. Aided Design*, 14:547–559, 1997.

[CBC+01] J. Carr, R. Beatson, J. Cherrie, T. Mitchell, T. Fright, B. McCallum, and T. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *Siggraph*. ACM, 2001.

[CDE+99] Siu-Wing Cheng, Tamal K. Dey, Herbert Edelsbrunner, Michael A. Facello, and Shang-Hua Teng. Sliver exudation. In *Proc. 15th Annu. ACM Sympos. Comput. Geom.*, pages 1–13, 1999.

[cga] The cgal library.

[Cha03] R. Chaine. A convection geometric-based approach to surface reconstruction. In *Symp. Geometry Processing*, 2003.

[CSD04] D. Cohen-Steiner and F. Da. A greedy delaunay based surface reconstruction algorithm. *The Visual Computer*, 2004. To appear.

[DG03] T. K. Dey and S. Goswami. Tight cocone: A water-tight surface reconstructor. *Journal of Computing and Information Science in Engineering*, 3:302–307, 2003.

[DG04] T.K. Dey and S. Goswami. Provable surface reconstruction from noisy samples. In *ACM SoCG*, 2004.

[Ede04] H. Edelsbrunner. Surface reconstruction by wrapping finite point sets in space. In B. Aronov, S. Basu, J. Pach, and M. Sharir, editors, *Ricky Pollack and Eli Goodman Festschrift*, pages 379–404. Springer-Verlag, 2004(?).

[EM90] H. Edelsbrunner and E. P. Mücke. Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms. *ACM Trans. Graph.*, 9(1):66–104, 1990.

[ES97] H. Edelsbrunner and N. R. Shah. Triangulating topological spaces. *Int. J. on Comp. Geom.*, 7:365–378, 1997.

[GJ02] J. Giesen and M. John. Surface reconstruction based on a dynamical system. In *Proceedings of the 23rd Annual Conference of the European Association for Computer Graphics (Eurographics), Computer Graphics Forum 21*, pages 363–371, 2002.

[GJ03] J. Giesen and M. John. The flow complex: A data structure for geometric modeling. In *ACM SODA*, 2003.

[GK00] P. Giblin and B. Kimia. A formal classification of 3d medial axis points and their local geometry. In *Computer Vision and Pattern Recognition*, Hilton Head, South Carolina, USA, 2000.

[GKS00] M. Gopi, S. Krishnan, and C.T. Silva. Surface reconstruction based on lower dimensional localized delaunay triangulation. In *Eurographics*, 2000.

[HOF01] H.K.Zhao, S. Osher, and R. Fedkiw. Fast surface reconstruction using the level set method. In IEEE, editor, *IWorkshop on Variational and Level Set Methods*, 2001.

[HS02] H. Hiyoshi and K. Sugihara. Improving continuity of voronoi-based interpolation over delaunay spheres. *Comput. Geom.*, 22(1-3), 2002.

[MT02] J-M. Morvan and B. Thibert. Smooth surface and triangular mesh : Comparison of the area, the normals and the unfolding. In *ACM Symposium on Solid Modeling and Applications*, 2002.

[PB01] S. Petitjean and E. Boyer. Regular and non-regular point sets: Properties and reconstruction. *Comput. Geom. Theory Appl.*, 19:101–126, 2001.

[Sib81] R. Sibson. A brief description of natural neighbour interpolation. In Vic Barnet, editor, *Interpreting Multivariate Data*, pages 21–36. John Wiley & Sons, Chichester, 1981.

[Yom81] Y. Yomdin. On the general structure of a generic central set. *Compositio Math*, 43, 1981.

Contents

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399