Eurographics Symposium on Geometry Processing (2003)
L. Kobbelt, P. Schröder, H. Hoppe (Editors)

A geometric convection approach of 3-D reconstruction

Raphaëlle Chaine

Prisme Project, INRIA Sophia-Antipolis, France

Abstract

This paper introduces a fast and efficient algorithm for surface reconstruction. As many algorithms of this kind, it produces a piecewise linear approximation of a surface S from a finite, sufficiently dense, subset of its points. Originally, the starting point of this work does not come from the computational geometry field. It is inspired by an existing numerical scheme of surface convection developed by Zhao, Osher and Fedkiw. We have translated this scheme to make it depend on the geometry of the input data set only, and not on the precision of some grid around the surface. Our algorithm deforms a closed oriented pseudo-surface embedded in the 3D Delaunay triangulation of the sampled points, and the reconstructed surface consists of a set of oriented facets located in this 3D Delaunay triangulation. This paper provides an appropriate data structure to represent a pseudo-surface, together with operations that manage deformations and topological changes. The algorithm can handle surfaces with boundaries, surfaces of high genus and, unlike most of the other existing schemes, it does not involve a global heuristic. Its complexity is that of the 3D Delaunay triangulation of the points. We present some results of the method, which turns out to be efficient even on noisy input data.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

1. Introduction : the reconstruction problem

Given a set of points that lie on or near an unknown surface, we consider the problem of computing a piecewise linear approximation of this surface. The reconstruction problem has received considerable attention both in computer graphics and computational geometry (see the state of the art report by Mencl and Müller [1] for a good classification of the existing works). In computer graphics, the early work by Hoppe *et al* [2] proposes an implicit approximation of the surface to be reconstructed. Curless and Levoy [3] have presented a similar approach dedicated to range images. Works by Bernardini *et al* [4] and Gopi *et al* [5] are closer to computational geometry approaches which are more combinatorial. They output a set of facets from a geometric data structure such as the Delaunay triangulation of the points. Historically, the earliest works on reconstruction in computational geometry were the α-shapes of Edelsbrunner [6] and the sculpture algorithm by Boissonnat [7]. Later on, Amenta and Bern [8] have proposed the first algorithm (CRUST) with correctness guarantees under a given sampling condition. An improved version of this algorithm (COCONE) [9] has also been described. It is worth mentioning that some existing algorithms do not necessarily extract the reconstructed surface from the 3D Delaunay triangulation of the points. For example, the POWER-CRUST of Amenta et al [10] uses a power-diagram of the points. Another algorithm is that of Mencl [11] which produces a triangulated surface by filling the contours of an extension of the Euclidean minimum spanning tree of the points. Attene and Spagnuolo [1] also use this tree and the Gabriel graph of the points to define new tetrahedra removal operations for sculpture algorithms. An other interesting approach is that of Boissonnat and Cazals [12] as they use the Voronoi diagram of the points to produce an implicit version of the surface. Eventually, recent approaches of Giesen and John [13] and Edelsbrunner [14] introduce the notion of flow in computational geometry. The work presented in this paper is of that kind.

The paper is organized as follows. In the second section, we briefly report on the evolution model proposed by Zhao, Osher and Fedkiw and the variational formulation associated to it (2.1). Then, we focus on the “fast tagging (convection) algorithm” that they have introduced to manage the first linear term of their evolution equation (2.2). In the third section, we establish a few properties of the fast tagging algorithm (3.1), and we interrelate it with a new Delaunay-based computational geometry algorithm that we introduce (3.2). This algorithm can be seen as some kind of sculpting algorithm that does not only produce volumes and that is not based on a priority-queue (3.3). It deforms a closed oriented *pseudo*-surface using a set of geometric and topological operations that we carefully detail (3.4). The convection process corresponds to the first term of the evolution equation proposed by Zhao, Osher and Fedkiw only. We propose a possible extension of our algorithm, which does not explicitly reflects the second term of the above equation but that resolves undetected pockets (3.5). The fourth section focuses on geometric properties and further statements that that can be helpful for a deeper understanding of the presented work (4). Eventually, we present some experimental results and conclude.

2. Convection model

2.1. Convection model proposed by Zhao, Osher and Fedkiw

In [15], Zhao, Osher and Fedkiw propose a function E to measure the distance between a surface Γ and a set of points Σ. This global distance or energy can be seen as some weighted area of the surface, where each surface element is weighted by its distance to its closest point in the data set Σ.

$$E(\Gamma) = \left(\int_{x\in\Gamma} d^p(x)\, ds \right)^{1/p}, 1 \leq p \leq \infty$$

where $d(x)$ is the distance from $x \in \mathbb{R}^3$ to its closest point in Σ.

Once this functional energy is defined, Zhao, Osher and Fedkiw suggest that the reconstruction problem can be solved by determining a surface which minimizes the global distance function to the data set Σ. They propose a variational formulation and an evolution equation to construct this minimal surface by deformation of a good initial enclosing approximation of the surface. More details on the way they extend this model to implicit level set surfaces and the way they get a partial differential equation can be found in their article [15].

The evolution proposed runs a gradient descent of the energy function to be minimized. At each step, every point x of the surface $S(t)$ evolves towards the interior of the surface, along the normal direction to $S(t)$ at point x, with a displacement speed that is proportional to :

$$-\nabla d(x) \cdot \vec{n} + (d(x)K)/p$$

K denotes the mean curvature of the surface at x and $\vec{n}$ is the inner normal at x. The tension of the surface represented by the second term $(d(x)K)/p$ is not linear, so that the evolution process requires a huge number of steps before reaching its equilibrium. In 2D, a steady state of this evolution equation is a polygon which has $K = 0$ everywhere except at the input data points where $d = 0$. These steady states do not generalize to polyhedra in 3D.

The better the initial approximation of the surface, the more the non linear -time consuming- effect of the evolution model is counteracted. In this paper, we focus on the convection model used by Zhao, Osher and Fedkiw to construct a good initial approximation of the surface. This model is equivalent to taking into account the first term of the above equation only. The authors show that this convection model can be physically motivated, and they solve it with a so-called “fast tagging (convection) algorithm”. Given a flexible enclosing curve or surface Γ, Zhao, Osher and Fedkiw put it into a velocity field $-\nabla d(x)$ created at point x by the distance function to the data set. In this velocity field, points of a curve or a surface are attracted towards their closest point in the data set, except those which are at the same distance from two or more data points. A curve (resp. surface) enclosing an area (resp. volume) can locally be considered as an infinite source of points so that it does not split into points. At the equilibrium, it reaches a polygon (resp. a polyhedron), the vertices of which belong to the data set. This is equivalent to making each point of the surface evolve along the normal direction to $S(t)$ with a displacement speed corresponding to the first term $\nabla d(x) \cdot \vec{n}$. Each point of the resulting surface also satisfies the steady state equation : $\nabla d(x) \cdot \vec{n} = 0$

2.2. Fast tagging (convection) algorithm

The “fast tagging algorithm” [15] is a fast numerical scheme that differentiates the interior from the exterior of a convection resulting surface. Once the distance d has been calculated at each point of a regular grid, the algorithm tags each point of this grid as interior $(-)$, exterior $(+)$ or boundary $(\Diamond)$: starting from a bounding connected set of points tagged as exterior $(+)$ (e.g. the points of a bounding box), a priority queue is built with the set of tagged exterior points that are adjacent to untagged points.

Let x be the point of the priority queue with the largest distance value (x is popped) :

- If one untagged neighbor of x has a distance value larger than $d(x)$, then x is tagged as a boundary point $(\Diamond)$.
- If all the untagged neighbors of x have their distance values smaller than $d(x)$, they are tagged as exterior $(+)$ and pushed into the priority queue.

At the end of the algorithm, the remaining untagged points are the interior points $(-)$. Fig. 1 illustrates the result of the “fast tagging algorithm” in 2D. Zhao et al show that this algorithm converges and has a complexity $O(NlogN)$, where N denotes the size of the grid.

The “fast tagging algorithm” is a numerical scheme that is clearly driven by the geometry. In this paper, we present a geometric algorithm that produces a similar result with-

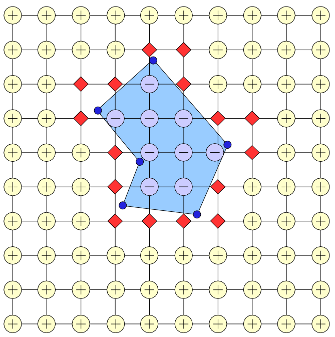

Figure 1: *"Fast tagging algorithm" : exterior points are tagged as $+$, interior points are tagged as $-$, boundary points are tagged as $\diamond$*

out using a grid. Related work has appeared in the computational geometry community. Recently, Giesen and John [13] have proposed a general study of the repulsion field $\nabla d(x)$, together with a reconstruction algorithm based on cells composed of points having the same attractor. Their work stresses the relationship between the considered cells and the Delaunay triangulation, but the reconstruction approach based on it cannot deal with surfaces with boundaries. A related idea has been developed by Herbert Edelsbrunner [14] that is also suitable for surface with boundaries. The reconstruction approach presented in our paper is similar. It has been developed independently, so that it is based on a different formalism and is more concerned with surface evolution and data structure issues.

3. A computational geometry approach to convection

3.1. Geometric properties

In this subsection, we establish geometric properties on the result of the convection process. More precisely, we want to bring out that the 2D parts resulting from the convection process (of an enclosing oriented surface towards a point set Σ) are composed of triangular oriented facets. These oriented facets are enclosed in the 3D Delaunay triangulation of Σ. We also show that they meet a particular geometric property. There is an equivalent of this result in the 2D case of curve convection in $\mathbb{R}^2$. We first present this simpler case to make a comprehensive step towards the 3D case.

Definition 1

- Given 2 points P_1 and P_2, the half-edge $\overset{\frown}{P_1P_2}$ denotes the oriented edge from P_1 to P_2. An edge can be seen as the union of 2 coupled half-edges.
- Given an edge e in $\mathbb{R}^2$, the diametrical disk of e is the union of 2 half-disks respectively supported by the two corresponding half-edges. The half-disk associated to a half-edge $\overset{\frown}{P_1P_2}$ is located on its left hand (see fig. 2).
- Given a set of points in $\mathbb{R}^2$, an edge is said to meet the Gabriel property, if its associated disk does not contain any point of the set. Similarly, we consider that a half-edge meets the Gabriel property if its associated half-disk is empty of sampled points.

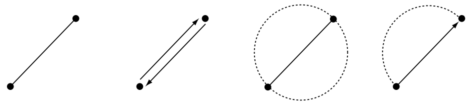

Figure 2: *$a)$ edge, $b)$ edge as the union of 2 coupled half-edges, $c)$ disk associated to an edge, $d)$ half-disk associated to a half-edge*

Lemma 2 Given a curve C enclosing a point set Σ in $\mathbb{R}^2$, the convection process of C towards Σ converges to a closed oriented *pseudo*-curve that is composed of a set of half-edges. These half-edges are enclosed in the 2D Delaunay triangulation of Σ and their associated half-disks are oriented towards the interior of the curve. Moreover, these half-edges meet the Gabriel property. In this context, the term *pseudo*-curve is used to mean that different parts of the evolving curve can locally adjoin.

Proof C is a closed bounding curve oriented towards the data. It can be considered as the union of several pieces of curve, so that each piece intersects $d = 2$ adjacent Voronoi cells only (see Fig. 3).

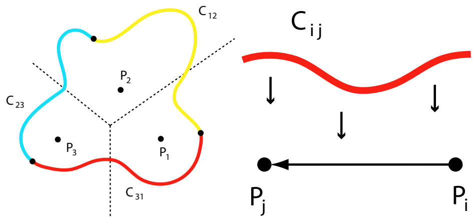

Figure 3: *Decomposition of C into pieces going through $d = 2$ adjacent Voronoi cells only*

Let C_{ij} be the piece of curve that intersects the Voronoi cells of P_i and P_j. Without loss of generality, we can suppose that C_{ij} and $\overset{\frown}{P_iP_j}$ are oriented consistently. Since the Voronoi cells of P_i and P_j are adjacent, the simplex P_iP_j is included in the Delaunay triangulation of the points. We consider the result of the convection of C_{ij} towards Σ. Along their inward normal vectors, the points of C_{ij} that lie inside the Voronoi cell of P_i (resp. P_j) are attracted to P_i (resp. P_j). The points equidistant from P_i and P_j are attracted to the middle of $[P_iP_j]$. This does not mean that moving points will eventually converge to their (local) attractor. During its evolution, the attractor of one point can change. Moreover, the points of C_{ij} do not evolve any more when they all meet $\nabla d(x) \cdot \vec{n} = 0$.

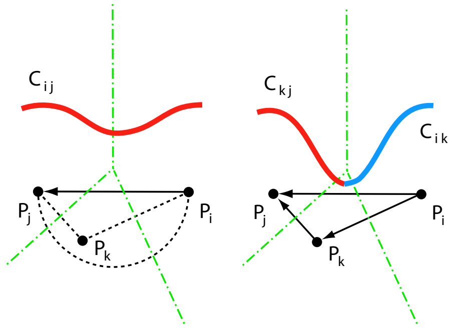

Figure 4: *Attraction towards a half-edge without the Gabriel property*

First case The half-edge $\widehat{P_iP_j}$ does not meet the Gabriel property (see Fig. 4). It means that its associated half-disk contains at least another point of the set. Let P_k be the point that lies in the half-plane associated to $\widehat{P_iP_j}$ and that is connected to P_i and P_j in the Delaunay triangulation [†]. P_k is one of the points included in the half-disk associated to $\widehat{P_iP_j}$ and the Voronoi center of P_i, P_j and P_k cannot be located in the half-plane delimited by $\widehat{P_iP_j}$ (consider the pencil of circles going through P_i and P_j). It implies that points of C_{ij} meet with the Voronoi cell of P_k on their way to their attractor. When this event occurs, C_{ij} can in turn be decomposed as the union of 2 pieces C_{ik} and C_{kj}, which intersect the adjacent Voronoi cells of P_i and P_k, and P_k and P_j respectively. The result of the evolution of C_{ij} is composed of the result of the evolution of C_{ik} and C_{kj}.

Second case The half-edge $\widehat{P_iP_j}$ meets the Gabriel property. In that case, no new Voronoi cell is encountered by the points of C_{ij} on their way to their attractor. The points of the curve are attracted by their closest point in the set and dragged by their neighbors on the curve, so that the result of the convection of C_{ij} is the entire half-edge $\widehat{P_iP_j}$, supported by an edge of the Delaunay triangulation.

Termination The above analysis dealt with the local evolution of the curve. There remains the eventuality of auto-intersections at different parts of the curve during the convection scheme. If ever the curve was to intersect itself at two coupled half-edges, then it would collapse in this area with a possible creation of a hole or even the creation of isolated points. Note that such cross-over do not happen in the case of a 2D convection (lemma 5 shows that each resulting half-edge has its entire diametrical disk empty : this indirectly implies that two coupled half-edges cannot be collapsed). At each step, the curve shrinks towards its interior, so that the positive area it encloses decreases. This ensures the termination of the convection scheme. □

Definition 3

- Given 3 points P_1, P_2 and P_3, the half-facet $P_1\widehat{P_2}P_3$ denotes the facet supported by P_1, P_2 and P_3 and oriented towards $\overrightarrow{P_1P_2} \wedge \overrightarrow{P_1P_3}$. A facet can be considered as the union of 2 coupled half-facets.
- Given a facet f in $\mathbb{R}^3$, the diametrical ball of f (smallest ball enclosing f) is the union of 2 half-balls respectively included in the 2 half-spaces defined by the 2 half-facets composing f.
- Given a set of points in $\mathbb{R}^3$, we extend the definition of the Gabriel property to facets that have their diametrical balls empty and to half-facets that have their associated half-ball empty.

Lemma 4 Given a surface S enclosing a point set Σ in $\mathbb{R}^3$, the 2D parts that are included in the convection result can be viewed as a set of closed oriented *pseudo*-surfaces. This lemma does not address the possibility to get 0-D parts (isolated points) nor 1-D parts. Each obtained *pseudo*-surface is composed of half-facets oriented inwards. These half-facets are embedded in the Delaunay triangulation of Σ and they all meet the Gabriel property. The term *pseudo*-surface means that different parts of the evolving surface can locally share common geometric information.

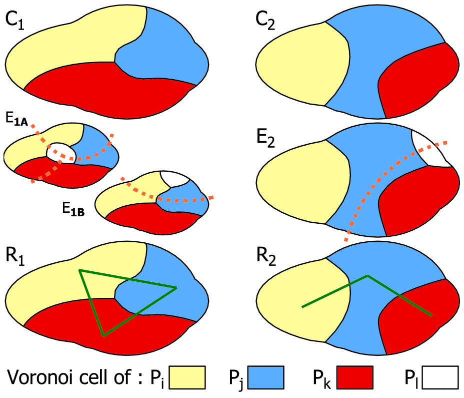

Figure 5: *Convection of a piece of surface S_{ijk}*

Proof S is a closed bounding surface oriented towards the data. It can be decomposed as the union of several pieces of surfaces, so that each piece S_{ijk} intersects the adjacent Voronoi cells of only 3 points P_i, P_j and P_k (S_{ijk} and $P_i\widehat{P_j}P_k$ being oriented consistently). The intersection of S_{ijk} with these 3 Voronoi cells can be of 2 kinds (configuration C_1 or C_2, see Fig. 5). During the convection process towards the interior, S_{ijk} can encounter the Voronoi cell of a new point

[†] We say that P_k (resp. $\widehat{P_iP_k}$ and $\widehat{P_kP_j}$) is the point (resp. are the half-edges) *hidden by* $\widehat{P_iP_j}$.

(event E_{1A} or E_{1B} on configuration C_1, event E_2 on configuration C_2, see Fig. 5). In that case, S_{ijk} can be decomposed in further pieces that intersect 3 adjacent Voronoi cells in turn (follow the dot line in Fig. 5). The result of the evolution of S_{ijk} is the union of the evolution of these new pieces. If S_{ijk} does not meet with a new Voronoi cell during the convection process, it converges either to $\overset{\frown}{P_iP_jP_k}$ (result R_1 for configuration C_1, see Fig. 5), either to the 1D edges P_iP_j and P_jP_k (result R_2 for configuration C_2, see Fig. 5), depending on the position of the circumcenter of P_i, P_j and P_k. In the case where S_{ijk} converges to $\overset{\frown}{P_iP_jP_k}$, the Gabriel property must be satisfied by $\overset{\frown}{P_iP_jP_k}$: otherwise the evolving S_{ijk} would have encountered the Voronoi cell of the point hidden by $\overset{\frown}{P_iP_jP_k}$.

Termination The above analysis deals with the local decomposition and evolution of S. The possibility of locally stopping this evolution because of global intersections at different parts of the evolving surface cannot be discarded in 3D (see Fig. 21). Half-facets resulting from the local evolution are embedded in the Delaunay triangulation so that they can intersect at Delaunay edges only. It implies that global cross-over occur at the level of (groups of adjacent) half-facets. The evolving surface is defined as the boundary of a volume so that an auto-intersection at the level of a half-facet implies the collapse of the current crossed parts. These cross-over of the surface can give raise to holes, changes in the number of connected components, but they can also create 1D parts (and perhaps 0D parts) included in the Delaunay triangulation. Along the convection process, the surface shrinks towards its interior so that the positive area it encloses decreases. This ensures the termination of the convection scheme. □

3.2. A computational geometric algorithm of convection

A computational geometric algorithm of convection can be derived easily from the previous results. Given a set of points Σ, the idea is to choose an enclosing surface embedded in the 3D Delaunay triangulation of the points and to make it evolve inside this tetrahedrization, by sculpting away some enclosed tetrahedra. More precisely, the algorithm we propose consists in making a closed oriented *pseudo*-surface shrink inside the 3D Delaunay triangulation of the points, until it locally fits Σ with half-facets satisfying the Gabriel property.

The evolving *pseudo*-surface S_{ev} is initialised with the convex hull of Σ and it is oriented inwards. Then, this oriented *pseudo*-surface S_{ev} evolves, subject to geometric and topological operations that ensure the connectivity restoration between half-facets, whenever a half-facet is shrunk. Before we detail these operations in subsection 3.4, the deformation scheme can roughly be described by the following algorithm :

SHRINK Algorithm

```
for each half-facet $\overset{\frown}{ABC}$ of the evolving pseudo-surface $S_{ev}$ do
    if $\overset{\frown}{ABC}$ does not meet the Gabriel property then
        if the coupled half-facets $\overset{\frown}{ABC}$ and $\overset{\frown}{BAC}$ both belong to $S_{ev}$ then
            - suppress $\overset{\frown}{ABC}$ and $\overset{\frown}{BAC}$ from $S_{ev}$
        else
            - replace $\overset{\frown}{ABC}$ with its 3 hidden half-facets $\overset{\frown}{ABD}$, $\overset{\frown}{BCD}$ and $\overset{\frown}{CAD}$
        end if
        - restore the connectivity between half-facets
    end if
end for
```

In this algorithm, the 3 hidden half-facets replacing a half-facet $\overset{\frown}{ABC}$ come from the Delaunay tetrahedron hidden by $\overset{\frown}{ABC}$. The case where the half-facet $\overset{\frown}{ABC}$ to be shrunk and its coupled half-facet $\overset{\frown}{BAC}$ both belong to S_{ev} corresponds to the case where the tetrahedron hidden by $\overset{\frown}{ABC}$ is outside the volume delimited by S_{ev}. Then, the opening of $\overset{\frown}{ABC}$ corresponds to a local auto-intersection with a local collapse of the *pseudo*-surface.

Fig. 6 illustrates the result of this algorithm in the 2D case. One can notice that the convection result can locally be composed of coupled half-edges that both meet the Gabriel property. We call them *thin parts*.

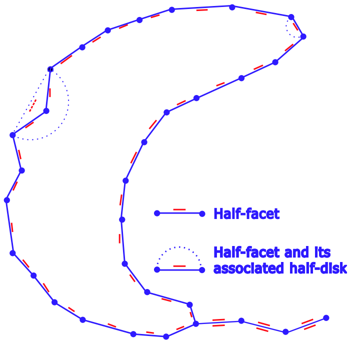

Figure 6: *Convection towards a 2D points set*

There are two versions of the SHRINK algorithm according to whether thin parts of the result are kept or not. We usually want to get rid of these non-manifold parts in case of volumes (see Fig. 7) but it is necessary to keep them in the case of surfaces with boundaries (or volume with thin parts). In this latter case, it is necessary to distinguish significant from undesirable thin parts. The most natural thing to do is to pursue and complete the convection process on these thin parts, starting from their boundary (a thin part boundary is an edge that is adjacent to (only) 2 coupled half-edges).

The algorithm extension proposed in section 3.5 was originally proposed to hollow undetected pockets out but is also efficient in detecting undesirable thin parts. Note that, in the case desirable thin parts are kept, it would be possible to retrieve a manifold by locally blowing some air in between coupled half-facets.

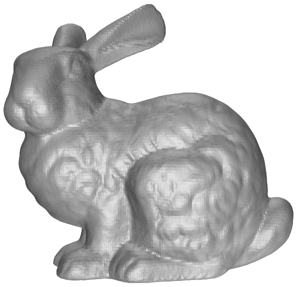

Figure 7: *Convection towards the Bunny point set, with removal of thin parts*

3.3. Comparison with the sculpture algorithm

The SHRINK algorithm can be seen as a generalization of the sculpture algorithm of Boissonnat [7]. Both approaches are quite different, however. In sculpture algorithms, a weight is given to each tetrahedron hidden by the evolving surface, so that each step of the algorithm results in the exudation of the tetrahedron with the largest weight. The weights chosen must favor the elimination of tetrahedra that lie behind larger, badly-shaped facets. Such an elimination process is run under control of a priority queue and under the constraint of topological genus invariance. The problem is that the order of the facets in the priority queue can be misleading. It does not take into account the global data configuration. In some cases, the sculpture process can locally be stopped to respect topological properties (see Fig. 8), while another elimination sequence could have driven to a better result. Veltkamp pointed out this problem in [16]. Other sculpting algorithms have been proposed to allow topological changes [17, 18], but they are still dependent on the operations order.

Unlike sculpture algorithms, the SHRINK algorithm is not based on a global heuristic and does not require a priority queue. The evolution of the triangulated surface is guided by a physical scheme and is not subject to topological conditions : the topological genus of the evolving surface can change several times before the process reaches the equilibrium.

3.4. Pseudo-surface : Data structure and operations

In subsection 3.2, we have presented an algorithm that makes an oriented *pseudo*-surface evolve, subject to topological and geometric operations. We now report on the data structure corresponding to a *pseudo*-surface and on the basic operations performed.

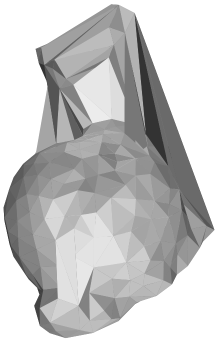

Figure 8: *Sculpture result : We have added topological constraints and a priority queue to turn our algorithm into a sculpture algorithm. The weight of a half-facet can be seen as some measure of the Gabriel property. The original shape could not be extracted because the algorithm got stuck by topological constraints.*

On a triangulated surface of a closed object, each facet lies on 3 different vertices and is adjacent to 3 different facets. Since the vertices of a triangulated surface are geometrically distinct, it is easy to retrieve the connectivity between facets from the geometry of its vertices.

This last remark does not hold for *pseudo*-surfaces as the ones produced by our algorithm, since different vertices, edges or half-facets can share common geometric information. The only way to get the connectivity information between half-facets is to store it into a suitable data structure, updated at each step of the convection process, thanks to a set of dedicated operations.

A *pseudo*-surface is a set of half-facets satisfying the following properties :

- each half-facet is adjacent to 3 other half-facets oriented consistently (2 or 3 of these half-facets can possibly be the same),
- each half-facet is incident to 3 different vertices,
- two different, non-adjacent vertices can share a common geometric information.

Such a *pseudo*-surface is a cellular complex, but not a simplicial complex : two adjacent half-facets can share more than one common edge. To retrieve an abstract simplicial complex, a barycentric subdivision (such as the one described by Vegter [23]) is required. After this operation, a half-facet is adjacent to 3 different half-facets, and each vertex is the center of a topological disk.

During the convection process, a *pseudo*-surface evolves so that a half-facet is opened to discover 3 new half-facets adjacent to a new vertex, or such that two coupled half-facets collapse.

It means that the data structure used to represent a *pseudo*-surface must support the following operations :

- replacement of a half-facet by its 3 hidden half-facets,
- collapse of 2 coupled half-facets (sharing the same geometry but oriented differently).

The operation of replacing a half-facet by its 3 hidden half-facets is similar to that of inserting a new vertex into a facet of a 2D triangulation [24]. This operation connects the 3 created half-facets with each other and with the half-facets adjacent to the deleted one. This is an operation that does not change the topology of the *pseudo*-surface. Note that the inserted vertex can possibly share common geometric information with another vertex of the *pseudo*-surface.

As far as the collapse of 2 coupled half-edges is concerned, there are 8 different configurations, depending on the number of common vertices and the number of common edges for the two coupled half-facets. Some of these operations modify the topological structure of the *pseudo*-surface. The collapse operations that involve at least one common edge are those to be used to pursue the convection process on thin parts.

3.4.1. 0 common vertex and 0 common edge

The simplest case is the one where the 2 coupled half-facets to collapse are not connected at all. The collapse of these two coupled half-facets involves 6 vertices and 6 edges. The restoration of the connectivity between the remaining half-facets implies the disappearing of 3 vertices and 3 edges : they are merged with the vertices and edges sharing the same geometry (see Fig. 9). This operation changes the topological structure of the *pseudo*-surface. It corresponds to a handle creation.

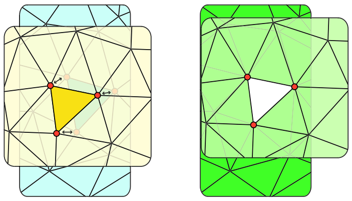

Figure 9: *0 common vertex and 0 common edge : disappearing of 3 vertices and of 3 edges*

3.4.2. 1 common vertex and 0 common edge

If the 2 coupled half-facets are only attached on a common vertex V, their collapse does not change the topological structure of the *pseudo*-surface (Euler characteristic unchanged). The 4 other vertices fusion by pair, whereas V is split into 2 vertices (see Fig. 10 –the dot pieces of circle represent sheets of the *pseudo*-surface–). 3 edges disappear by fusionning with edges sharing the same topology.

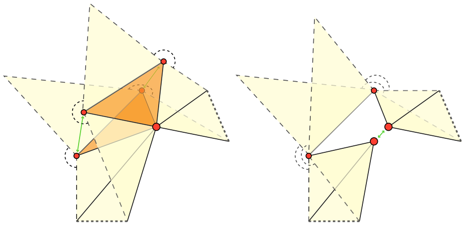

Figure 10: *1 common vertex and 0 common edge : disappearing of 2 vertices and creation of a new one, disappearing of 3 edges*

3.4.3. 2 common vertices and 0 common edge

If the 2 coupled half-facets are connected at 2 different vertices V_1 and V_2, the collapse operation changes the topological structure of the *pseudo*-surface, but that change can be of 2 sorts and cannot be characterized locally :

- either a creation of a new connected component,
- either the opening of a handle.

The collapse operation corresponds to the fusion of the unconnected vertices and to the split of V_1 and V_2 (see Fig. 11). The 6 six edges fusion by pair.

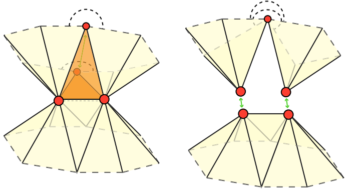

Figure 11: *2 common vertices and 0 common edge : disappearing of a vertex and creation of 2 others, disappearing of 3 edges*

3.4.4. 3 common vertices and 0 common edge

The collapse of 2 coupled half-facets sharing the same vertices can yield 4 sorts of modifications in the topological structure of the *pseudo*-surface (the problem of determining which of this change occur is global) :

- either a creation of 2 new connected components,
- either the opening of a handle, and the creation of a new connected component,
- either the opening of 2 handles,
- either the opening of 3 handles, and the creation of a new one.

As far as the connectivity restoration between half-facets is concerned, the 3 common vertices are split and the 3 pairs of edges are merged (see Fig. 12).

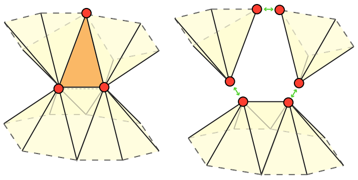

Figure 12: *3 common vertices and 0 common edge : creation of 3 new vertices, disappearing of 3 edges*

3.4.5. 2 common vertices and 1 common edge

When the 2 coupled half-facets share a common edge *e*, they necessarily share the vertices at the extremity of this edge. This collapse operation does not change the topological structure of the *pseudo*-surface. The edges different than *e* and the 2 vertices that are not adjacent to *e* are fusionned by pairs (see Fig. 13). *e* disappears with the 2 coupled half-facets.

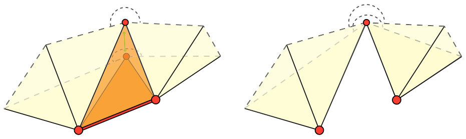

Figure 13: *2 common vertices and 1 common edge : disappearing of 1 vertex, disappearing of 3 edges*

3.4.6. 3 common vertices and 1 common edge

In the case where the 2 coupled half-facets share a common edge *e* but also their last vertex V, the collapse operation changes the topological structure of the *pseudo*-surface, with the creation of a new connected component or the opening of a handle.

The vertex V is split and the pair of edges that are different than *e* are fusionned (see Fig. 14). *e* disappears.

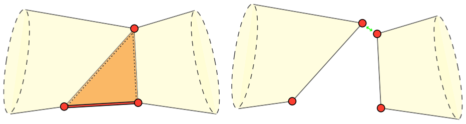

Figure 14: *3 common vertices and 1 common edge : creation of 1 vertex, disappearing of 3 edges*

3.4.7. 3 common vertices and 2 common edges

Two coupled half-facets that share 2 common edges necessarily share all their vertices. The collapse operation does not change the topological structure of the *pseudo*-surface in that case. The two common edges disappear with the vertex they share. The pair of remaining edge is merged (see Fig. 15).

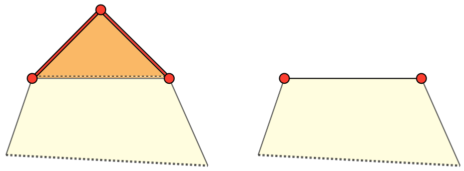

Figure 15: *3 common vertices and 2 common edges : disappearing of 1 vertex, disappearing of 3 edges*

3.4.8. 3 common vertices and 3 common edges

The last collapse case occurs when the 2 coupled half-facets share all their vertices and all their edges : all these vertices and edges disappear with the half-facets (see Fig. 16)

This operation changes the topological structure of the *pseudo*-surface, with the disappearing of a connected component.

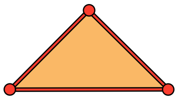

Figure 16: *3 common vertices : disappearing of 3 vertices and 3 edges*

3.5. Extension of the convection process

Zhao, Osher and Fedkiw use the convection process to initialise another process that minimizes an energy function. Unfortunately, the convection process can be stuck by the presence of important concavities. These are larger than all the hidden half-balls that can be raised from the resultant interface between the cavity and the outside (Fig. 17 illustrates it in 2D). Such cavities are denoted pockets by Edelsbrunner et al [20].

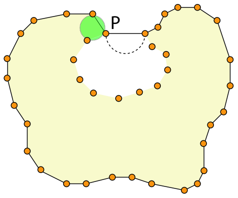

Figure 17: *No digging is possible, because of a concavity larger than the half-ball around its hiding half-facet*

Let us see how our scheme could be improved in order to solve this issue. If the surface is locally sampled finely enough to reflect the presence of the cavity, the distance from the points outside the cavity to their neighbors inside the cavity should be bounded by a factor of $\mathbf{lfs}(P)$, (even if this factor is not always as fine as desired). It means that one can expect some constraint of consistency between selected half-facets and local density, even if the sampling conditions are not verified everywhere (at the level of a crease edge, for instance).

Let P be a point of a facet blocking a cavity. If the cavity is sufficiently sampled to be detectable, the distance from P to its surface neighbors, on and outside the cavity, should be small with respect to the size of the blocking facets (see Fig. 17). Furthermore, the distance from P to its surface neighbors can be reflected by the local density of points.

Consequently, we have improved the opening condition, so that a half-facet that meets the Gabriel property can also be shrunk if its size is not coherent with the local 3D density of the sampling (which is also indicative of the distance to the skeleton). The size of a half-facet is reflected by the radius of its associated half-ball and the problem remains to find a good approximation of the density. In practice, we have approximated the point density around a point by the distance to its fourth nearest point, keeping in mind that the mean number of neighbors in a 2D triangulation is 6. One can argue that it is not a solid solution, but we think that a better approximation with theoretical coefficients and thresholds could be computed from the recent works by Erickson [21] and by Boissonnat and Attali [22].

An other extension of the algorithm can be made in order to reconstruct surface of objects with internal holes. Suppose that the input data points belong to two unconnected concentric spheres, one can argue that the shrinking algorithm reconstructs the outer sphere only. The result of the presented SHRINK algorithm clearly depends on the part of the object surface that is reachable by the initial enclosing surface. However, it is possible to repeat the shrinking process with the set of unreached points, and so on.

4. Oriented nature of the opening condition

This section can be skipped on a first reading. It contains statements that can help gaining a better understanding of computational geometry reconstruction algorithms. It also illustrate the difficulty to translate a concept as “the restrained Delaunay triangulation” of a surface into a discrete equivalent.

In the convection algorithm presented above, the evolving *pseudo*-surface is shrunk through a half-facet if its associated half-ball is not empty. This means that the half-facets included in the result have empty associated half-balls, but not necessarily empty diametrical balls. This also means that the convection process is locally driven by the internal skeleton only. In this section, we study how the convection algorithm evolves if we replace the opening condition by an orientation-free version of it.

2D point sets

Lemma 5 In the case of convection towards a 2D point set, it does not change the result to keep the original opening condition or to enlarge it to half-edges whose entire diametrical disk contains a point. A consequence of that result is that the evolving curve cannot intersect itself and collapse at two coupled half-edges.

Proof It is equivalent to prove that each half-edge resulting from the current oriented algorithm has its entire diametrical disk empty. Suppose that $\overset{\frown}{P_jP_i}$ is a half-edge of the evolving curve (or of the result) whose coupled half-edge $\overset{\frown}{P_iP_j}$ does not meet the Gabriel property (see Fig. 18). Let P_h be the point hidden by $\overset{\frown}{P_iP_j}$. By construction, one of the half-edges $\overset{\frown}{P_hP_i}$ or $\overset{\frown}{P_jP_h}$ must have been inserted in the evolving oriented curve at a previous step of the algorithm. This facet did not meet the Gabriel property, so that it could be shrunk to discover $\overset{\frown}{P_jP_i}$. This is geometrically incompatible with $\overset{\frown}{P_iP_j}$ not meeting the Gabriel property. □

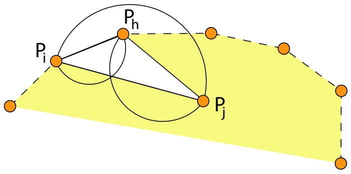

Figure 18: *2D convection property : A discovered half-edge $\overset{\frown}{P_jP_i}$ must have its coupled half-edge meet the Gabriel property also, otherwise it could not have been inserted in the evolving surface*

3D point set A well-known result by Amenta and Bern [8] ensures that the global shape of a smooth object surface can be retrieved from a set of points, when these points form an ε-sample of the original surface, with $\varepsilon < 0.1$. It means that the distance between any surface point P and the closest sampled point is less than ε times the distance $\mathbf{lfs}(P)$ to the medial axis. Under this condition, the “restricted Delaunay triangulation” constitutes a piecewise linear approximation of the surface, homeomorphic to it. It is composed of Delaunay facets whose dual Voronoi edges intersect the surface ($\varepsilon < 0.1$ ensures that the number of intersections between Voronoi edges and the original surface is less than 1). In practice, it is sometimes impossible to get an ε-sample — consider, for instance, the case of surfaces that are locally not differentiable — and precise reconstruction of crease edges still remains a problem.

Petitjean and Boyer [19] have proposed a discrete generalization of the ε-sample notion, to deal with scattered data considered independently of any surface. In this generalization, a set of points is a discrete ε-sample if :

- a triangulated surface can be built from it,

- each facet of this triangulation has size (*granularity*) less than ε times the distance from its vertices to their furthest Voronoi vertices.

The facets of such a triangulation belong to the 3D Delaunay triangulation of the points and they all meet the Gabriel property. Given a discrete ε-sample with $\varepsilon < 1$, the authors also propose an algorithm which constructs a surface as a set of Gabriel facets in the 3D triangulation of the points. In practice, point sets rarely meet the conditions required by Petitjean and Boyer. Their algorithm most often produces surfaces with boundaries, even when the original surface is unbounded. It is the case for the "Bunny" point set. To illustrate that, we ran our algorithm with a different opening condition on the half-facets : a half-facet is shrunk if its entire diametrical ball contains a point. Fig. 19 shows the result of the contraction process after the suppression of the thin parts. As one can see, there were not enough Gabriel facets to stop the progression and the folding up of the surface into itself. This can be a problem if one wishes to get a closed surface.

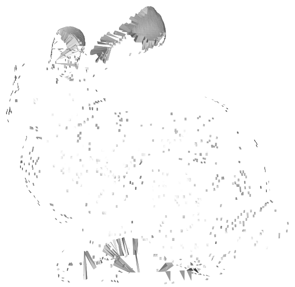

Figure 19: *Bunny : Shrinking of half-facets whose entire associated ball contains a point (thin parts are removed). This illustrates the convection result when the oriented nature of the half-facet opening condition is suppressed.*

Why, in practice, are there so many half-facets of the result that meet the Gabriel property but whose entire diametrical ball is not empty ? The point is that the result available in 2D cannot be extended to the 3D case, and it is easy to extract configurations of the data where a discovered half-facet has its associated half-ball empty, but not its entire diametrical ball :

Let $\overset{\frown}{ABC}$ be a half-facet of the surface. The half-ball associated to $\overset{\frown}{ABC}$ (let us say *in front of* $\overset{\frown}{ABC}$) is empty. The half-ball associated to $\overset{\frown}{BAC}$ (let us say *behind* $\overset{\frown}{ABC}$) can contain a point if the Delaunay tetrahedron $ABCD$ behind $\overset{\frown}{ABC}$ is of the following kind : the circumcenter O of $ABCD$ does not lie in the half-space associated to $\overset{\frown}{ADC}$ (so that $\overset{\frown}{ADC}$ can be shrunk to discover $\overset{\frown}{ABC}$), nor in the half-space associated to $\overset{\frown}{BAC}$ (see Fig. 21).

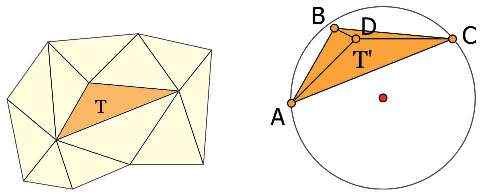

Figure 20: *In front of and behind a half-facet $T = \overset{\frown}{ABC}$*

Suppose that a Delaunay tetrahedron such as the one described on the above figure is present among a set of points corresponding to a well sampled surface. If the surface is a surface with boundary and that both the half-facets $\overset{\frown}{ADC}$ and $\overset{\frown}{BAC}$ can be reached by the convection process, a cross-over in the convection process can occur with a possible creation of a handle if none of the edges AB, BC and CA are boundaries (see Fig. 21).

Set of points :
-1 1 0; -1 0 0; -0.5 0 0
1 1 0; 1 0 0; 0 0.4 -0.05
0 1 0; 0 -1 0; 0.5 0 0
-1 -1 0; 1 -1 0; 0 0.4 0.05

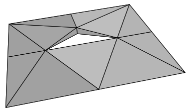

Figure 21: *Convection towards a 3D point set belonging to a nearly planar surface ($\mathbf{lfs}(P) = \infty$). A cross-over occurred due to the presence of boundaries and a sliver*

5. Results

An implementation of the convection algorithm has been done using CGAL [24]. All the results presented here have been obtained in a few seconds on a Linux platform (Intel 4 CPU 2.00GHz 1993 MHz, 768 MBytes RAM,9GBytes DISK space).

6. Conclusion

In this paper, we have dealt with an existing physical convection scheme, that we have translated into a geometric algorithm. This approach seems interesting to us, both to fasten the usual discretization processes and to extract the possibly underlying geometric structure of an evolution equation. This geometric algorithm is based on the 3D Delaunay triangulation of the points, but one can imagine an implementation where the latter does not need to be constructed : not all the Delaunay facets are explored and a convex hull algorithm

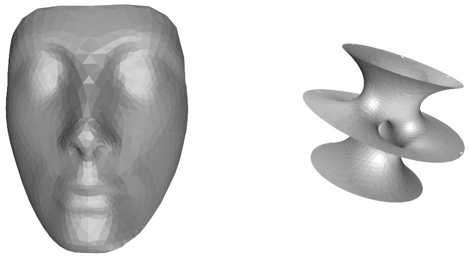

Figure 22: *"Nefertiti" (surface with boundaries, 1128 points, time 0.170 s) and "Mathematical surface"(surface with boundaries, 6752 points, time 1.100s)*

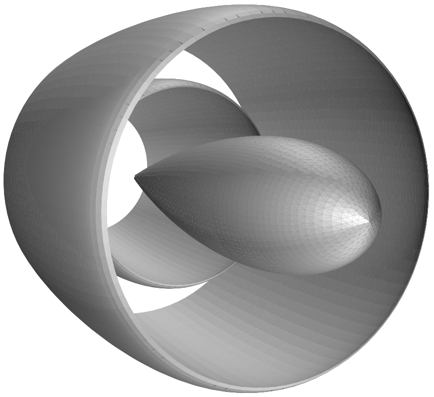

Figure 23: *"Plane Engine" (surface with boundaries, 11444 points, time 4.210s)*

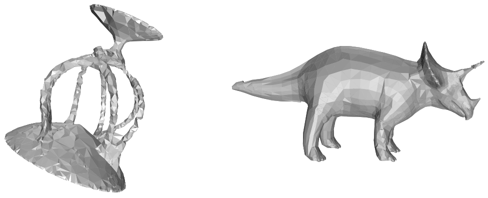

Figure 24: *"Schale" (surface with boundaries, 2714 points, time 0.399s) and "Triceratops" (surface without boundary, 2833 points, time 0.498s)*

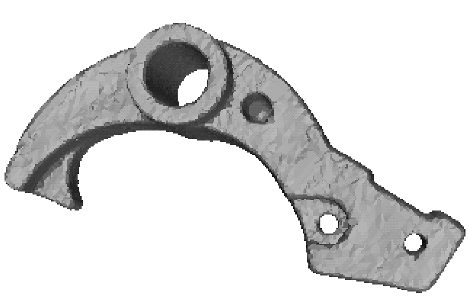

Figure 25: *"Meca" (surface without boundary, 12594 points, time 2.638s)*

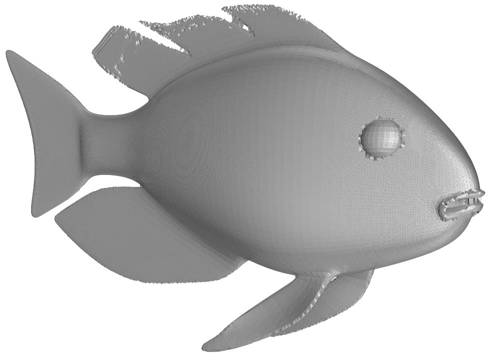

Figure 26: *"Fish" (surface with thin parts, 54811 points, time 10.128s)*

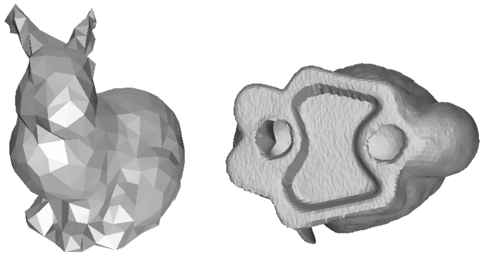

Figure 27: *Other aspects of "Bunny" : undersampled data, holes below the Bunny basis (surface with open boundaries)*

coupled with a location data structure could be enough to determine local density at a point and to determine if a half-facet encounters the Gabriel property or not. The current results of the algorithm could also be improved with a better approximation of the point density on the surface. Our convection algorithm compute an oriented *pseudo*-surface but a post-processing is possible to convert it into a manifold triangular mesh.

Acknowledgements

Work partially supported by the IST Programme of the EU as a Shared-cost RTD (FET Open) Project IST-2000-26473 (ECG - Effective Computational Geometry for Curves and Surfaces) and by I3S laboratory (UMR 6070 CNRS-UNSA). Thank to Pierre-Marie Gandoin, Steve Oudot and all those who had a close look to this article and the ideas it contains. Their reactions, encouragements and remarks are priceless.

References

1. Robert Mencl and Heinrich Müller. Interpolation and approximation of surfaces from three-dimensional scattered data points. In *State of the Art Reports, Eurographics*, pages 51–67, 1998.
2. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *Comput. Graphics*, 26(2):71–78, 1992. Proc. SIGGRAPH '92.
3. B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proc. SIGGRAPH 96*, pages 303–312, 1996.
4. F. Bernardini, C. Bajaj, J. Chen, and D. Schikore. Triangulation-based object reconstruction methods. In *Proc. 13th Annu. ACM Sympos. Comput. Geom.*, pages 481–484, 1997.
5. M. Gopi, S. Krishnan, and C. T. Silva. Surface reconstruction based on lower dimensional localized Delaunay triangulation. In *Eurographics*, 2000.
6. H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Trans. Inform. Theory*, IT-29:551–559, 1983.
7. Jean-Daniel Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Trans. Graph.*, 3(4):266–286, 1984.
8. Nina Amenta and Marshall Bern. Surface reconstruction by Voronoi filtering. *Discrete Comput. Geom.*, 22(4):481–504, 1999.
9. 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.
10. 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.
11. Robert Mencl. Surface reconstruction from unorganized points in space. In *Abstracts 11th European Workshop Comput. Geom.*, pages 67–70. Universität Linz, 1995.
12. 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.
13. Joachim Giesen and Matthias John. Surface reconstruction based on a dynamical system. In *Proc. Eurographics*, 2002.
14. Herbert Edelsbrunner. Surface reconstruction by wrapping finite sets in space. Technical report, to appear in Ricky Pollack and Eli Goodman Festschrift, ed. B. Aronov, S. Basu, J. Pach and M. Sharir, Springer-Verlag, 2002.
15. H.K.Zhao, S. Osher, and R. Fedkiw. Fast surface reconstruction using the level set method. In *Proceedings of IEEE Workshop on Variational and Level Set Methods in Computer Vision (VLSM)*, 2001.
16. Remco C. Veltkamp. Boundaries through scattered points of unknown density. *Graphical Models and Image Processing*, 57(6):441–452, 1995.
17. Leila De Floriani, Paola Magillo, and Enricco Puppo. Managing the level of detail in 3d shape reconstruction and representation. In *Proc. 14th International Conference on Pattern Recognition (ICPR'98), Brisbane, Australia*, volume 1, pages 389–391, 1998.
18. M. Attene and M. Spagnuolo. Automatic surface reconstruction from point sets in space. In *Computer Graphics Forum*, volume 19, august 2000.
19. Sylvain Petitjean and Edmond Boyer. Regular and non-regular point sets : properties and reconstruction. *Computational Geometry : Theory and applications - Elsevier*, 19(2-3):101–126, 2001.
20. M.A Facello H. Edelsbrunner and J.Liang. On the definition and the construction of pockets in macromolecules. *Discrete Appl. Math*, 88:83–102, 1998.
21. Jeff Erickson. Nice point sets can have nasty Delaunay triangulations. In *Proc. 17th Annu. ACM Sympos. Comput. Geom.*, pages 96–105, 2001.
22. Dominique Attali and Jean-Daniel Boissonnat. Complexity of the Delaunay triangulation of points on polyhedral surfaces. In *Proc. 7th ACM Symposium on Solid Modeling and Applications*, 2002.
23. Gert Vegter. *Handbook of Discrete and Computational Geometry - J.E. Goodman and J. O'Rourke eds*, chapter 28 - Computational topology, pages 517–536. CRC Press LLC, Boca Raton, Florida, 1997.
24. *http://www.cgal.org*.