A Stochastic Approach for Multiresolution of Solid Objects with Topological Control

Esdras Medeiros, [a] Luiz Velho [a,*]

[a] *IMPA–Instituto de Matemática Pura e Aplicada*

Helio Lopes [b] Thomas Lewiner [b]

[b] *PUC–Pontifícia Universidade Católica do Rio de Janeiro*

Abstract

In this article we introduced a filtering method, by means of topological operators, for a representation in multiresolution of a solid object. In each level of resolution, we have a sampling by Poisson disks with particular characteristics. With this filtering, it is possible to control the resolution and topology changes in a unified way, through the stellar and handle operators.

Key words: Poisson disc sampling, solid topology

1 Introduction

Solid objects, in our context, are compact manifolds with the same dimension of the surrounding space. The frontier is the most significant part because it determines both the geometric shape and the topology. Topology, in a brief way, may be understood as the number of holes and connected components in a solid object. Its main property is the invariance by homeomorphisms.

Our research work on solid objects relates four subjects that are well known to the scientific community in Mathematics and in Computer Graphics. They are: sampling, reconstruction, modeling in multiresolution and topology.

Sampling and reconstruction of Graphical Objects (1) are maxims in computer graphics. Thanks to the Shannon sampling theorem (5), signals may be fully

* Luiz Velho
Email address: `lvelho@impa.br` (Luiz Velho).

represented by samples uniformly spaced with a sampling rate lower than half of the bandwidth.

In the context of surfaces we find equivalent results in the works by Amenta et al. (2) and F. Bernardini et al. (3). These papers determine sampling conditions that assure the existence of a simplicial reconstruction which is topologically equivalent to the original surface. In (3) the recostruction is characterized by an *alpha solid*. Since we are dealing here with solid objects we will obtain a similar result for them by means of a reconstruction based on *solid alpha complexes* (6). We will see ahead that a solid alpha complex is closely related to the alpha solid. More precisely, the boundary of a solid alpha complex is an alpha solid.

We emphasize that differently from the Shannon theorem related to signals where the reconstruction is a exact one, the reconstruction of a surface or a solid object is represented by a simplicial complex, that is, only an approximation. Actually, the "exact" term is associated to the topological equivalence property.

The scale spaces have as the fundamental idea to model a signal $f : \mathbb{R}^d \to \mathbb{R}$ in a family of different scales as $L : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$, where L is a transformation defined as the convolution of f with the Gaussian kernel. This transformation works as a low pass filter. As we reduce the variance of the Gaussian kernel we also reduce the bandwidth of the transformed signal, allowing the reduction of the sampling rate required to the reconstruction of it. Therefore, we may represent a scale space as a family of samplings with increasing rates. The reconstruction of the family elements generate a sequence of infinite signals that, in the limit, converges to the original signal.

What we will propose along the next pages is something similar to representations of signals in scale spaces but in the context of solid objects. The most challenge we find for it is to express "topological" details that can be detected in the different scale levels. Fortunataly, we achived a nice result thanks to the represantations by a *solid alpha complex* family, in different rates, of a solid object's samplings. As we increase the sampling rate, we are able to identify two types of details that are emphasized: the geometrical details characterized by the shape of the holes and a detail characterized by the quantity of holes. The last one is what we called "topological" detail. In this article we will see a model of multiresolution representation of solid objects that treats these two types of details in a unified framework of topological operators.

1.1 Related Work

There are many works related to representations of simplicial meshes in different scales. In the works by Kobbelt (9) and Velho et al. (10), the several resolution levels are generated by means of subdivision rules. In the work by Puppo (12) a study is performed on simplicial multiresolution of planar domains. In the work by Hoppe (13) we find representations of meshes that progressively change resolution using *edge collapse* operations. The limitations of these mentioned works, besides of other similar ones, is that the topology is already previously determined by the represented object (or by the base mesh) in all levels of resolution. In short, there is no change of topology.

Talking about the topological treatment in a family of simplicial complexes, the work by Edelsbrunner et al. (14) is one of the works that is more similar to ours. They define *topological persistence* by means of a *topological filtering*. Their results have several applications. Among them we point out the analysis of topological noise in protein samplings and the simplification of morse-smale complexes. One gap of this article is that there is no approach related to multiresolution.

1.2 Contributions

Our work has as contribution, to unify the simplicial representations of bidimensional solid objects in different scale levels and put some control in the change of topology between them.

We deal with multiresolution samplings and reconstructions of solid objects within a stochastic approach. To achieve that, we use samplings with Poisson disks distribution because they allow us a strong control of the scale. It also give us the interesting property that its reconstruction by means of solid alpha complexes has a bounded aspect ratio on the triangles. This last characteristic implies in the fact that we have a good quality in the triangulated mesh.

The main result we obtain in this article is an algorithm (with two versions) that generates a sequence operators (11) by refining or simplification. The topological operators may be of two types: the *stellar* operators which change the resolution and combinatorial structure of the mesh and the *handle* operators which change the topology. As a consequense, the sequence of operators generated by these algorithms links the resolution levels independently of their different underlying topologies.

1.3 Summary

The article is written as follows. On section 2 we introduce and revisit concepts that will be used along the text. Among them we point out the simplicial concepts, nested family of points, Delaunay triangulation, solid alpha complex and Voronoi diagram. We also establish some notations. On section 3 we present the data structure adopted to represent the triangular meshes and we survey the topological operators that will be used to insert and to remove points in a mesh. These operators will allow the interaction in the diffrent resolution levels. On section 4 we submit the concepts of samplings by Poisson disks, characterize the dual complex and exhibit conditions to have a reconstruction topologically equivalent to the original solid object. In section 5 we establish several other significant concepts about scaling family in order to generate a filtering by topological operators. In one of the main theorems of this section we state that the sequence of operations generated by the refining algorithm is inverse to the sequence of operations generated by the simplification algorithm. We proceed to extensions of previous results in order to cover a wider class of samplings to allow us to achieve results which are more practical in terms of implementation and mesh quality. In section 6 we present a conclusion of the work and list the future work.

2 Preliminary Concepts and Notations

In this section we will describe some basic concepts in topology and in computational geometry that will be used in this work. We will also introduce the *Solid Alpha Complex* concept.

2.1 Simplicial Complexes

A k-simplex $\sigma_T = conv(T)$ is the convex combination of a set of points linearly independent $T \subset \mathbb{R}^n$, $\#T = k+1$; $0 \leq k \leq n$; where # denotes cardinality. k is the dimension of simplex σ_T.

Definition 1. *A* Simplicial Complex K *is a collection of simplices that satisfy the following properties:*

(1) If $\sigma_T \in K$ then $\sigma_U \in K$, if $U \subset T$. We say that σ_T is the face of σ_U.
(2) If $\sigma_U, \sigma_V \in K$, then $\sigma_{T\cap V} = \sigma_U \cap \sigma_V$.

The two properties above imply that $\sigma_{T\cap V} \in K$. We will name $|K|$ as the subspace of $\mathbb{R}^n$ covered by K. A *subcomplex* L of K is a simplicial complex such that $L \subset K$. The dimension of a simplicial complex K is given by the widest dimension among all simplices of K. We will represent by K^i the subset of the simplices of K with dimension i.

A *Solid Simplicial Complex* does not have isolated simplexes, i.e., k-simplexes that are not faces of a simplex of wider dimension. Given a simplicial complex K, the collection $\overline{K} \subset K$ is the maximal solid simplicial complex contained in K. We also named such complexes as *meshes*.

In a simplicial complex K of dimension n we say that a simplex of dimension $n-1$ is *boundary* if it is a face of only one simplex of dimension n. The boundary operator ∂ of a simplicial complex K is such that $\partial K = \{\ \sigma \in K$ where σ is boundary or is a face of a boundary simplex$\}$. Observe that ∂K is also a simplicial complex and $\partial K = \partial \overline{K}$.

Let $S \subset \mathbb{R}^n$ be a finite set of points and K a solid simplicial complex. If $K^0 \subset S$, by simplicity we will write (S, K) as a notation (read "pair points S and simplicial complex K" or simply "pair"). The collection $\mathcal{F} = \{S_i\}_{i \in \{1,2,\ldots n\}}$ is a *nested family of points* if $S_1 \subset S_2 ... \subset S_n$. To each of the i's we call *level*. Then lets define a *nested family of meshes*:
Definition 2. *Let $\mathcal{F} = \{S_i\}_{i \in \{1,2,\ldots n\}}$ be a nested family of points. We say that $\mathcal{M}(\mathcal{F}) = \{(S_i, K_i)\}_{i \in \{1,2,\ldots n\}}$ is a Nested Family of Meshes.*

Notice that in a nested family of meshes, the meshes have common vertices, however, they can increase or reduce the quantity of vertices as i increases. We will see ahead that this last definition is fundamental to formalize structures in multiresolution with changes of topology.

Based on the structure of simplicial complexes we will define some simplicial artifacts such as Delaunay Triangulations, Alpha Complexes and Alpha Shapes.

2.2 Triangulation and Voronoi Diagram

Delaunay Triangulation of a set of points in the plane is the unique set of triangles that connect such points and that satisfy the property of the "empty circle": the circunscribed circle of each triangle does not include any other point. In a certain sense, it is the most natural manner to triangulate a set of points. Below we will give a general definition based in simplicial complexes.
Definition 3. *Given a set $S \subset \mathbb{R}^n$ in general position, the Delaunay triangulation of S is the simplicial complex $\mathrm{DT}(S)$ that comprises only*

(1) all the k-simplices, σ_T $(0 \le k \le n)$, with $T \subset S$ such that the circumsphere (the smallest sphere, such that all points are in its boundary) of T

does not contain no other point of S, and

(2) all the k-simplices that are faces of other simplices are also in $\mathrm{DT}(S)$.

We will define the Voronoi diagram and will establish its relation with Delaunay triangulation.

Definition 4. *Let S be a set of n points in the plane. for each $s \in S$, the Voronoi region $V(s)$ is a set of points of the plane closest to p than other points of S. The Voronoi diagram $V(S)$ is the partition of the plane generated by the regions of Voronoi of S.*

We have then, the proposition below, which is well known in the literature.

Proposition 1. *The Delaunay triangulation of S is the dual graph of the Voronoi S diagram: two points of S are linked by an edge in the triangulation of the Delaunay triangulation if and only if its regions of Voronoi are incidents in the diagram of Voronoi of S.*

Proof. See (17). □

2.3 Alpha Complexes and Alpha Shapes

Alpha Complexes are simplicial complexes that describe levels of detail of cluster points. Through the variation of a real positive number α we obtain different shapes, from the more refined to the more coarse. The more refined is the set of points itself, achieved when $\alpha = 0$. As α increases, the shape also increases by the addition of simplices developing cavities that may gather or split. The coarser form is the Delaunay triangulation which is obtained for great values of α. More precisely, the Alpha Complexes have the following definition:

Definition 5. *Let $S \subset \mathbb{R}^n$ be a set of points in a general position. For $T \subset S$ with $\#T \leq n$, let b_T and μ_T be the smallest ball that contains points of T and its radius, respectively. Given $0 \leq \alpha \leq \infty$, the alpha complex $\mathcal{C}_\alpha(S)$ of S is the sub complex of* $\mathrm{DT}(S)$ *where the simplex $\sigma_T \in \mathrm{DT}(S)$ is in $\mathcal{C}_\alpha(S)$ if:*

(1) $\mu_T < \alpha$ and $b_T \cap S = \emptyset$, or

(2) σ_T is a face of other simplex in $\mathcal{C}_\alpha(S)$.

Observing the definitions of the Delaunay triangulation and Alpha Complex, the following properties are immediate:

P1. If $\alpha_1 \leq \alpha_2$ then $\mathcal{C}_{\alpha_1} \subset \mathcal{C}_{\alpha_2}$,

P2. $\mathcal{C}_\alpha \subset \mathrm{DT}(S)$, $\forall \alpha > 0$ and

P3. $\mathcal{C}_\infty = \mathrm{DT}(S)$.

The alpha shapes S_α is defined as $|C_\alpha(S)|$. Thus, as in alpha complexes we

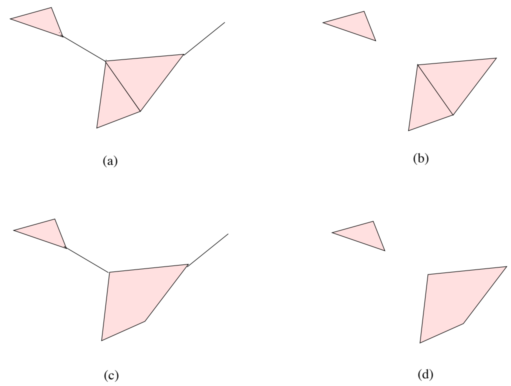


Fig. 1. The alfa complex (a) and its solid alpha complex (b). The *alpha shape* (c) and alpha solid (d).

obtain a Delaunay triangulation for great α parameters also in alpha shapes we achieve precisely the convex hull. In fact, an alpha shape is adequate for the generalization of the concept of convex hull, being well adopted in many applications. See for instance (16).

2.4 Alpha Solid and Solid Alpha Complex

In general, the alpha complex and the alpha shape are respectively simplicial complexes and polytopes composed by simplexes of different dimensions. Bernardini et al.(4) defined the *solid alpha shape* (or simply alpha solid) as the *alpha shape* without isolated k-simplices. In a similar way we define the Solid Alpha Complex as the solid alpha without k-simplexes isolated. It is a type of subcomplex which is a "regularized" version of the alpha complex. As we saw above, it is a maximal solid simplicial complex.

In figure 1 we show the difference between the alpha complex and the solid alpha complex in the 2D case. We will denote the solid alpha complex of a set $S \subset \mathbb{R}^n$, given $0 \leq \alpha \leq \infty$, as $\overline{\mathcal{C}_\alpha(S)}$. Notice that the properties P1, P2 e P3 are still valid for solids alpha complexes.

3 Topological Operators and Mesh Representations

3.1 Topological Operators

Now we will introduce an unified framework of basic operations in manifolds of dimension two with or without boundary. There are two types of operators over meshes: handlebody operators that change the topological characteristic and stellar operators that change the resolution and the combinatorial structure. We will present them under the computational point of view, i.e., as API's, and for a full explanation of the mathematical theory see the work by Velho et. al(11). Our main objective is to use these operators to link the different levels of a nested family of meshes. In a more precise way, the jump of a certain level to a neighbor level will be a sequence of applications of these operators.

Our first operator is $\mathsf{add(s)}$ which returns s, where $s \in S$. It will act only over points of S in some pair (S, K). Its functionality is to add the vertex s to the set S. In a similar way we have $\mathsf{remove(s)}$ as the operator that removes the vertex s of S and returns such vertex. These operators are an extension of the traditional operators that we will see ahead. They supply more flexibility to change the quantity of points between the levels without changing the mesh structure.

The *handlebody* operators allow us to cut and paste pieces of surfaces. They are:

- $\mathsf{f} = \mathsf{create(p0,\ p1,\ p2)}$: creates a new face f from the points $\mathsf{p0}$, $\mathsf{p1}$ and $\mathsf{p2}$ of S;
- $\mathsf{(p0,\ p1,\ p2)} = \mathsf{destroy(f)}$: destroys an existing face and return its three points;
- $\mathsf{e} = \mathsf{glue(he0, he1)}$: "identifies" two boundary half-edges and turns them into an interior edge which is returned;
- $\mathsf{(he1,\ he2)} = \mathsf{unglue(e)}$: divides an interior edge in two boundaries and returns them.

The star operators allow us change the resolution (quantity of the points in the mesh) and the mesh combinatorial structure. They are:

- $\mathsf{e} = \mathsf{flip(e)}$: makes a swap in the edge e and returns the same edge. Notice that the flip operator is defined only for internal edges;
- $\mathsf{v} = \mathsf{split(f)}$: trisects the face f and returns a new vertex which is added to S;
- $\mathsf{f} = \mathsf{weld(v)}$: an inverse operator of the split operator, it returns one face and removes a vertex from S.

Let Δ be a topological operator and consider the pair (S, K). We denote by $\Delta(S, K)$ as the *action* of Δ over (S, K) that also generates a new pair (S', K').

In all, we have nine topological operators over pairs and we can clearly observe that they are invertible. More precisely, we say that Δ^{-1} is the inverse of Δ if $\Delta^{-1} \circ \Delta = \Delta \circ \Delta^{-1} = Id$, where Id is the identity operator, that is, $Id(S, K) = (S, K)$. All topological operators are invertible. Thus, we have: $\mathsf{add}^{-1} = \mathsf{remove}$, $\mathsf{destroy}^{-1} = \mathsf{create}$, $\mathsf{glue}^{-1} = \mathsf{unglue}$, $\mathsf{flip}^{-1} = \mathsf{flip}$ e $\mathsf{split}^{-1} = \mathsf{weld}$.

There is no commutative property between operators. In fact, if we take as an example $\mathsf{split} \circ \mathsf{create(p0, p1, p2)}$, there is no way the operator split commute with create because of the dependence between arguments. In this case, it is required that the face pre-exists to the split operator actuate.

Among the topological operators, only two of them increase the resolution of a mesh: create and split. The difference between them is that create operator somtimes have a depedence on the vertices that are in S and needs to be preceded of *add* operator. This does not take place with split operator that only have a dependence on a face that should be in K. We can reason in the same way when there is a decrease of resolution.

The definitions below formalize the topological *junction* and *disjunction* operations over nested family of meshes.

Definition 6. *Let (S_0, K_0) and (S_1, K_1) be two distincts pairs. We say that $(S_1, K_1) = (S_0, K_0) \oplus s$ is a topological junction of a point s with (S_0, K_0) if $S_1 = S_0 \cup \{s\}$ and there exists a sequence of topological operations Δ_i, such that*

$$\Delta_n \circ \Delta_{n-1} ... \circ \Delta_1(S_0, K_0) = (S_1, K_1).$$

Definition 7. *Let (S_0, K_0) and (S_1, K_1) be two distincts pairs. We say that $(S_1, K_1) = (S_0, K_0) \ominus s$ is a topological disjunction of a point s with (S_0, K_0) if $S_1 = S_0 - \{s\}$ and there exists a sequence of topological operations Δ_i such that*

$$\Delta_n \circ \Delta_{n-1} ... \circ \Delta_1(S_0, K_0) = (S_1, K_1).$$

It follows, directly from the operators invertibility that the topological junction and disjunction are also invertible. More precisely:

Property 1. $(S_1, K_1) = (S_0, K_0) \oplus s \Rightarrow (S_0, K_0) = (S_1, K_1) \ominus s$. *In another way:*

$$\Delta_n ... \circ \Delta_1(S_0, K_0) = (S_1, K_1) \Rightarrow \Delta_1^{-1} ... \circ \Delta_n^{-1}(S_1, K_1) = (S_0, K_0).$$

From the non commutativity of operators it follows that the order of topological junction of points influences directly the mesh final result.

3.2 Mesh Representation

We will follow the same data structure by Velho et. al (11) to represent 2D meshes. This structure has the advantage of unifying the functionalities of the topological operators that will be seen on section 4. We will rewrite them here.

A mesh is structured as $M = (V, E, F, B)$ where V, E, F, B are collections of vertices, edges, faces and boundary curves, respectively.

```
struct Surface {
  Container<Face*> faces;
  Container<Edge*> edges;
  Container<Vertex*> vertices;
  Container<Edge*> bndries;
}
```

The face stores a pointer for the first half-edge of the internal cycle.

```
struct Face {
  Half_Edge* he;
}
```

A edge is formed by two half-edges. If it is representing a boundary edge, one of the half-edges points to a null face.

```
struct Edge {
  Half_Edge he[2];
}
```

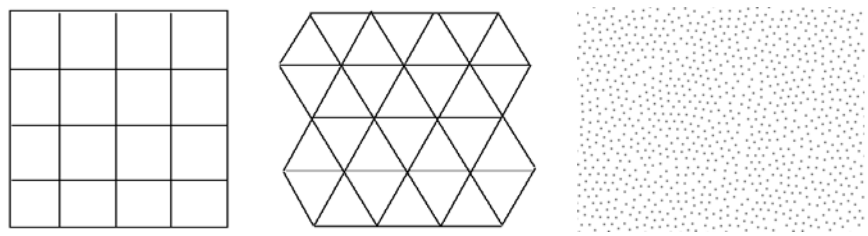
The half-edge is the core element of the data structure. It stores a pointer for its initial vertex, a pointer for the next half-edge in the cycle of the face and pointers for the edge and face to which it belongs. Notice that the mate half-edge may be accessed by the parent edge pointer.

```
struct Half_Edge {
  Vertex* org;
  Half_Edge* next;
  Face* f;
  Edge* e;
}
```

The vertex stores a pointer for the incident half-edge.

```
struct Vertex {
  Half_Edge* star_i;
  Data d;
}
```

In the collection of boundary curves, the representative of each element is an edge that belongs to such curve.



(a) (b) (c)

Fig. 2. PDS's examples: (a) quadrilateral lattice, (b) triangular lattice, (c) stochastic sampling.

4 Solid Sampling

In this section we will treat the Poisson Disks Sampling (PDS), an important class of stochastic sampling often used in Computer Graphics applications. For learning reasons we will define PDS's in two types of domains: all plane $\mathbb{R}^2$ and regions $R \in \mathbb{R}^2$ that are open, limited and connected. We will also analyze their geometrical and topological relation with the solid alpha complexes.

4.1 PDS's in $\mathbb{R}^2$

Definition 8. *Let $S_\alpha = \{s_1, s_2, ...\}$ be a sampling in the plane. We say that S_α is a Poisson Disks Sampling if (i) $\cup_{s_i \in S_\alpha} B_\alpha(S_i) = \mathbb{R}^2$ and, additionally, (ii)$S_\alpha \cap B_\alpha(s_i) = \{s_i\}$, $\forall i$. The condition (i) will be named covering condition and condition (ii) wil be named Poisson condition.*
Proposition 2. *There exists a PDS in the plane.*

Proof. Trivial examples of PDS's are the regular *lattices* as the quadrilateral (figure 2.a) and the triangular (figure 2.b). More complexes examples may be created by means of the dart throwing algorithm approach *dart throwing* (19). In this approach, we have a random generator of samples in a given region and a validator that checks if they satisfy the geometric criteria expected. In our case we are considering the whole plane as the sampling region and the the Poisson condition as geometric criteria. If a sample is validated, then it is incorporated to the output, if not, we discharge it. The algorithm interruption criteria would be the coverage condition, which, logically, is not feasible because there is an infinity number of points to be sampled in the plane. Therefore, it is only applicable for limited domains as we will define soon. This algorithm is a typical example of stochastic sampling also known as blue noise (20). See, for instance figure 2.c. □

As follows, we will establish the relation between an PDS and its solid alpha

complex
Proposition 3. *The Solid Complex Alpha of a PDS S_α of the plane, $\overline{C_\alpha(S_\alpha)}$, is a coverage of the plane.*

Proof. It is sufficient to show that every triangle $\sigma \in DT(S_\alpha)$ satisfies $r_\sigma < \alpha$ where r_σ is the circumscribed circle radius of σ. Let c_σ be the center of the circumscribed circle of σ. If, as an absurd, $r_\sigma > \alpha$ then directly from the Delaunay property we would have $d(c_\sigma, S_\alpha) > \alpha$ and, therefore, $c_\sigma \notin \cup_{s_i \in S_\alpha} B_\alpha(s_i)$. □

The proposition above has as consequence, two important facts. The first one is the quality of the planar subdivision in triangles. There is an upper bound for the aspect-ratio$(\sigma) = \frac{L^2}{vol(\sigma)}$ where $vol(\sigma)$ is the triangle area of σ and L is the length of the longest edge of σ. In Medeiros et al. (7) work it is evidenced that $\frac{L^2}{vol(\sigma)} \leq 4\sqrt{3}$ and the equality takes place when the wider angle is $2\pi/3$.

The second consequence is the scale control power of the simplicial elements. It is easy to see that by the sampling conditions, the radius of the circunscribed circles of the triangles and the edges lengths are variating in the interval $[\frac{\alpha}{2}, \alpha]$. From there, it is natural to think in multiresolution to represent the samplings of solid regions.

4.2 PDS's in Regions of the Plane

We will define the sampling by Poisson disks for a class regions of the plane with boundary. The frontier has a crucial importance: it defines both the shape and the topology of the region.
Definition 9. *Let $S_\alpha = \{s_1, s_2, ..., s_n\}$ be a sampling of the solid region R ($R = A \cup \partial A$, A limited, open and connected). We say that S_α is a Poisson disks sampling (PDS) if $R \subset \cup_{s_i \in S_\alpha} B_\alpha(s_i)$ (covering condition) and, additionaly, $S_\alpha \cap B_\alpha(s_i) = \{s_i\}$, $\forall i$ (Poisson condition).*

Observe that differently from the previous definition, the equality does not hold in the coverage condition. However, depending on the sampling radius [1] we can approximate the covering to the region as much as intended. We will see this fact more detailed in proposition 7.

As in the plane, there are PDS's for regions. With the same idea of generating a PDS using the dart throwing algorithm we have also the existence of an infinity of them.

[1] We will always be referring to the parameter α of Poisson disks sampling.

From here on, whenever we refer to a solid R region, it will be open, connected and limited unless the contrary stated.

In the definition below, each sample of a PDS of a region R may be classified in accordance to its topology.

Definition 10. *Let S be a sampling of a region R. We say that $s \in S_\alpha$ is boundary (interior) if $s \in \partial R$ ($s \in int(R)$).*

Talking about the techniques to generate a PDS of a region we may classify them as *approximating* and *interpolating*. The only difference between these two samplings is that the interpolating take the boundary into consideration.

Definition 11. *Let $S_\alpha = \{s_1, s_2, ..., s_n\}$ be a sampling of R. We say that S_α is an interpolanting sampling if exists $A \subset S_\alpha$ such that $A \subset \partial R$ and $\partial R \subset \cup_{s_i \in A} B_\alpha(s_i)$. We will denote an interpolating PDS by $\tilde{S}_\alpha$.*

Notice that using an adtaptation of the dart throwing by performing it in two searated steps we can easily obtain an interpolating PDS sampling. In the first step restrict the target sampling to the boundary of the region and later sample the subregion defined by its interior minus the covering of the disks in the.

Definition 9 lead us to questions about shape approximations and topology equivalence related to the region R and its solid alpha complex recostruction over some PDS. To enunciate and answer them first we will state some significant definitions that will serve as tools to express and to demonstrate the results.

Definition 12. *The weighted squared distance of one point $x \in \mathbb{R}^2$ from one ball b is given by $\pi_b(x) = ||x - c_b||^2 - r_b^2$ where c_b and r_b are the center and the radius of b, respectivelly.*

An important observation is that a point $x \in \mathbb{R}^2$ belongs to a ball if and only if $\pi_b(x) \leq 0$, and it belongs to the boundary of the ball if and only if $\pi_b(x) = 0$. Given a finit set of balls $\mathcal{B}$, we can podemos divide the space into regions:

Definition 13. *The Voronoi region of a ball $u \in S$ is the set of points of the plan for which u minimizes the weighted distance,*

$$V_u = \{x \in \mathbb{R}^2 | \pi_u(x) \leq \pi_v(x), \forall v \in \mathcal{B}\}$$

The diagram comprising the Voronoi regions is called, in the literature, a *power diagram*. It is not difficult to show that the set of points equally distant from two balls u and v is a hyperplane defined by $\pi_u = \pi_v$. In the context where we are, since we consider the disks of an PDS, we have balls with the same radius, and, therefore, the power diagram coincides with the Voronoi diagram. Observe that the Voronoi potential regions decompose the union of balls of one PDS in convex regions of type $V \cap u$, as it is illustrated in figure 4.

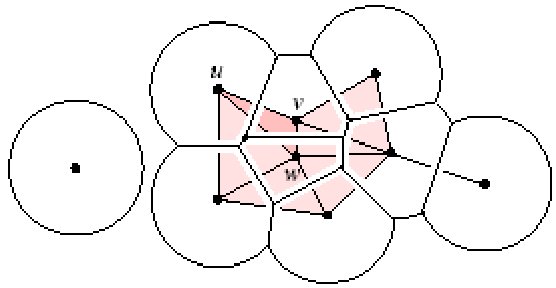


Fig. 3. Union of nine balls, convex decomposition using power Voronoi regions and its dual complex.

Definition 14. *The dual complex K of $\mathcal{S}$ is a collection of simplices*

$$K = \{conv\{u_c | u \in T\} | T \subseteq S, \cap_{u \in T}(u \cap V_u) \neq \emptyset\}.$$

Proposition 4. *The dual complex of a PDS S_α is an alpha complex.*

Proof. See (15). □

We have then, a different manner of defining the alpha complex of a set of points associated to spheres, differently from what was defined in 5. Depending on the objective to be achieved, one can use the more adequate definition.

Now, we will define some tools that will allow us to arrive to a conclusion on the topology of the alpha complexes. Actually, what we want to discover is what type of topological relation exists between the union of the balls of an PDS and its dual alpha complex.

Definition 15. *A deformation retraction of a space $\mathbb{X}$ onto a subspace $\mathbb{A}$ is a family of maps $f_t : \mathbb{X} \to \mathbb{A}, t \in [0,1]$ such that f_0 is the identity map, $f_1(\mathbb{X}) = \mathbb{A}$ and $f_t|\mathbb{A}$ is the identity, for all t. The family should be continuous, in such a way that the associated map $\mathbb{X} \times [0,1] \to \mathbb{X}$, $(x,t) \mapsto f_t(x)$ is continuous.*

In other words, starting from the original space $\mathbb{X}$ in time 0, we continuously deform the space to transform it in subspace $\mathbb{A}$ on time 1. A retraction deformation is a particular case of homotopy.

Definition 16. *A homotopy is a family of maps $f_t : \mathbb{X} \to \mathbb{Y}, t \in [0,1]$, such that its associated map $F : \mathbb{X} \times [0,1] \to \mathbb{Y}$ given by $F(x,t) = f_t(x)$ is continuous. Then , $f_0, f_1 : \mathbb{X} \to \mathbb{Y}$ are homotopic via the homotopy f_t. We denote this as $f_0 \simeq f_1$.*

Let us suppose that we have a retraction as in definition 15. we consider $i : \mathbb{A} \to \mathbb{X}$ an inclusion, we have that $f_1 \circ i \simeq id$ and $i \circ f_1 \simeq id$. This will allows us to classify $\mathbb{X}$ and its subspace $\mathbb{A}$ as having the same connectivity. This is a special case of homotopic equivalence.

Definition 17. *A map $f : \mathbb{X} \to \mathbb{Y}$ is called a homotopy equivalence if there is a map $g : \mathbb{Y} \to \mathbb{X}$, such that $f \circ g \simeq id$ and $g \circ f \simeq id$. Then, $\mathbb{X}$ and $\mathbb{Y}$ are homotpy equivalent and have the same homotopy type. This fact is denoted as $\mathbb{X} \simeq \mathbb{Y}$.*

Now we can enunciate an important result about homotopy between two spaces we have knowledge of.

Proposition 5. *Let S_α be a PDS of a region R and $\mathcal{B} = \cup_{s_i \in S_\alpha} B_\alpha(s_i)$. Then $\mathcal{B} \simeq |C_\alpha(S_\alpha)|$.*

Proof. It is not difficult to see that the idea of demonstration is to exhibit a retraction that takes space $\mathcal{B}$ in space $|C_\alpha(S_\alpha)|$. See in (15) for more details. □

The relevance of the proposition 5 is in the invariance of the homology between the two spaces. This type of relation does not preserve the intrinsic dimension once the alpha complexes may have isolated simplexes of smaller dimensions. However, there is a relation stronger which is the homeomorphism and in this case there is a coincidence of topology.

Definition 18. *A homeomorphism $f : \mathbb{X} \to \mathbb{Y}$ is a bijection, such that f and f^{-1} are continuous. We say that $\mathbb{X}$ is homeomorphic in relation to $\mathbb{Y}$, $\mathbb{X} \approx \mathbb{Y}$ and that $\mathbb{X}$ and $\mathbb{Y}$ have the same topology type.*

To have an assurance of topological equivalence between the region and the solid alpha complex, we will use the idea of a medial axis and LFS - local feature size.

Definition 19. *The medial axis of a curve F is the closing of a set of points in the plane that has two or more closest points in F.*

Definition 20. *The Local Feature Size, $LFS(p)$, of a point $p \in F$ is the euclidian distance of p to the nearest point m from the medial axis.*

In the proposition below we point out a condition for PDS's of solid regions, with smooth boundaries so that the dual complex has the same type of topology of the region.

[2]

Proposition 6. *If the radius of a PDS S_α (with smooth boundary) is lower than $\frac{1}{2} inf_{p \in \partial R} LFS(p)$ then $|\overline{C_\alpha(S_\alpha)}| \approx R$.*

Proof. See Medeiros (8). □

[2] This condition is to assure that exists $\epsilon > 0$ such that $LFS(p) > \epsilon$, $\forall p \in \partial R$.

About the geometric proximity of ∂R and $\partial C_\alpha(S_\alpha)$ we can establish a global convergence from the parameter α.

Proposition 7. *Let $\alpha > 0$ and S_α be a PDS of R. The Hausdorff distance $d_H(R, \partial|C_\alpha(S_\alpha)|)$ of region R with the region covered by the alpha complex of S_α is lower than α.*

Proof. It is sufficient to notice that if $\forall p \in R$ then there exists $q \in S_\alpha$ such that $p \in B_\alpha(q)$ therefore $d_H(p, |C_\alpha(S_\alpha)|) < \alpha)$. $\square$

In the literature, the algorithms to calculate alpha complexes, generally use the Delaunay triangulation in an intermediate stage. However, having in hands the proposition 6 we can use the ball-pivoting algorithm restricted to the plane (RBPA) (6) to build the solid alpha complex in linear time on the number of points without being required to previously calculate the Delaunay triangulation.

Proposition 8. *Ler $S \subset \mathbb{R}^2$ be a set of points in general position and $\overline{C_\alpha(S)}$ its solid alpha complex. Consider $\mathcal{T}_\alpha$ as the output of RBPA being performed in the plane. Then $\overline{C_\alpha(S)} = \mathcal{T}_\alpha$.*

Proof. See Medeiros et al. (6). $\square$

Therefore, given α as in the proposition 6, by the previous proposition, we calculate the solid alpha complex with the same type of topology of the region.

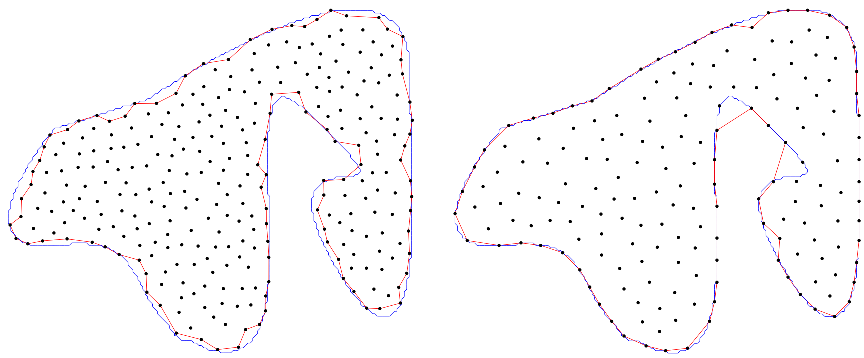
As it was previously seen, there are restrictions on the set of solid objects for which exists such $\alpha > 0$. In fact, when we demand that the boundary region be smooth it is because we do not want sharp features since then we would havepois $\inf_{p\in\partial R} LFS(p) = 0$ there is no manner to determine an upper bound for α. We are working to extend the assurances of reconstruction in proposition 6 to regions with a finite set of sharp features.

An important fact to be observed is that the dual complexes of approximating PDS's are noisier in the frontier than the dual complexes of interpolating PDS's (see figure 4). We will explain this fact better under section 6.

4.3 Boundary Approximation

Lets define when a α-pair $(\tilde{S_\alpha}, K_\alpha)$ is a *good* approximation of a region, reminding that the symbol ˜ means to say that S_α is an interpolating sampling.

Definition 21. *Let $(\tilde{S_\alpha}, K_\alpha)$ be a α-pair of a region R. Consider $P = \partial\overline{K_\alpha}$. If all vertices of P are boundary then we say that K_α is a good approximation of R.*



(a) (b)

Fig. 4. Examples of reconstructions of an approximating sampling (a) and an interpolating sampling (b).

As discussed above, from the dual complexes, the approximating PDS's are noisier on the boundary than the interpolating PDS's since it does not take points which are representatives of the boundary. Consequently, when parameter α tends to zero the sequence given by a family of solids alpha complexes of interpolating PDS's that are good approximations assure convergence of the normals, which does not take place with approximating PDS's families. Therefore, in geometrical terms the interpolating samplings that are good approximations are better.

Not always a α-pair $(\tilde{S_\alpha}, \overline{C_\alpha(S_\alpha)})$ is a *good* approximation. In figure 4.b we note that the interior point lies in the boundary of the reconstruction. In order to turn arounf this problem, lets define a Quasi-Alpha Complex of dimension 2 with with the purpose of ensuring that it has characteristics combined of the definitions of a Solid Alpha Complex and of a good approximation.

Definition 22. *Let S be any sampling with topological informations in the points (boundary or interior) of a region R. Given $\alpha > 0$, we denote $QC_\alpha(S)$, as a Quasi-Solid Alpha Complex (QSAC), the solid simplicial complex of S that satisfies the following properties:*

(1) $QC_\alpha(S) \subset Del(S)$

(2) $\sigma_{(s_i,s_j,s_k)} \in QC_\alpha(S)^2 \iff$

  (a) or $\mu_\sigma < \alpha$, *where* μ_σ *is the circumscribed radius of* σ*;*

  (b) or $\{s_i, s_j, s_k\} \cap int(R) \neq \emptyset$*;*

  (c) or $\exists$ B_l *e* B_m, *distinct connected components of the boundary* R, *such that* $B_l \cap \{s_i, s_j, s_k\} \neq \emptyset$ *and* $B_m \cap \{s_i, s_j, s_k\} \neq \emptyset$*;*

Notice that by property 2(a) it follows that $\overline{C_\alpha(S)} \subset QC_\alpha(S)$. From defition above, we have the following lemma:

Lemma 1. *If S is an interpolant PDS of a region R then:*

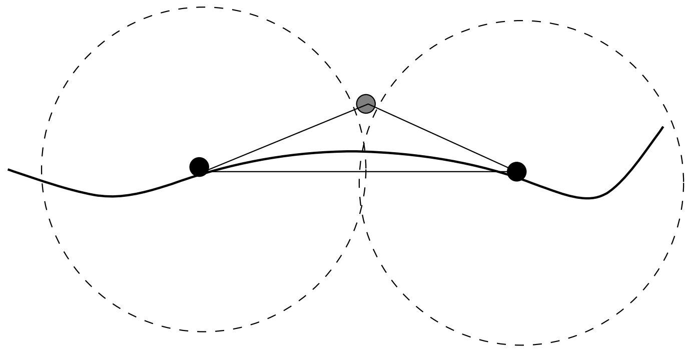


Fig. 5. In this example the light gray point is in the interior and is close to the boundary. The plotted triangle may have a very wide aspect ratio.

(1) $QC_\alpha(S)$ is a good approximation;
(2) If B_l is a connected component of the boundary of $QC_\alpha(S)$ then all its points lies in the same connected component of the boundary of R.

Proof. The item (1) follows directly from property 2.b of definition 22. The item (2) is a direct consequence of property 2.c in definition 22. $\square$

As we wish, a α-par $(\tilde{S_\alpha}, QC_\alpha(\tilde{S_\alpha}))$ is always a good approximation and therefore assures the convergence of normal. We also have that $QC_\alpha(\tilde{S_\alpha})$ have important properties of being a subset of the Delaunay triangulation and super set of the solid alpha complex. The disadvantage results from the fact that they do not have a superior limit for the aspect ratio of the set of triangles. It is what we will try to solve in the next section.

4.4 The Aspect Ratio

The aspect ratio with no upper limit takes place because interior points sampled too close to the boundary eliminate, in their neighborhood, the possibility of sampling boundary points that would be more representatives (see figure 5). To solve this, we need a sampling strategy to assure that interior points are not sampled in a determined neighborhood $\epsilon(\alpha)$ of the boundary region. When we tried to adopt this idea we had to use a weaker version of PDS's that are (α, β)-ADP's.

Definition 23. *Let $S_{\alpha\beta} = \{s_1, s_2, ..., s_n\}$ be a sampling of a solid region R ($R = A \cup \partial A$, A open, connected and limited) and $0 < \beta \leq \alpha$. We say that $S_{\alpha\beta}$ is a (α, β)-PDS if (i)$R \subset \cup_{s_i \in S_{\alpha\beta}} B_\alpha(s_i)$ (covering condition) and, beyond, (i)$S_{\alpha\beta} \cap B_\beta(s_i) = \{s_i\}$, $\forall i$ (Poisson condition).*

Notice that when parameter β approaches α, the sampling is more similar to a PDS. The scale notion is implicit in parameter β. As follows we will exhibit a way to generate a (α, β)-PDS of a region assuring that $\beta = \alpha/2$.

Proposition 9. *Given α, we can generate a $(\alpha, \alpha/2)$-PDS of a region R from the following steps:*

(1) $I = \{$ PDS of $R - \cup_{s \in \partial R} B_\alpha(s)\}$;
(2) $B = \{$ PDS of $\partial R\}$;
(3) $P = \{$ "PDS" of $R - \cup_{s \in B \cup I} B_\alpha(s)$ with conditional projection in the boundary$\}$. The conditional projection is made after a the generation anda validation of a sampling s. If $dist(s, \partial R) < \frac{\alpha}{2}$ then select s' as the nearest point of ∂R, else, select s.

Proof. By construction it is clear that $R \subset \cup_{s \in I \cup B \cup P} B_\alpha(s)$. We will show that $\beta = \alpha/2$. Let s' be the projection of s in ∂R. Let $p \in \partial R$ be the sampling point nearest to s. We know that $ss' < \alpha/2$ and that $ps > \alpha$. By the triangular inequality, we have that $ps' + ss' > ps \Rightarrow ps' > ps - ss' > \alpha - \alpha/2 > \alpha$. Then $\beta = \alpha/2$. □

Observe that $s \in int(R)$ and from steps 2 and 3 we conclude that $dist(s, \partial R) > \alpha/2$. Then, we can say that there exists a neighbor $\epsilon(\alpha) = \alpha/2$ of the boundary that does contain interior boundary points. As mentioned before this improve the problem of existence of thin triangles, i. e., great aspect ratio along the boundary of an α-par $(\tilde{S_\alpha}, QC_\alpha(\tilde{S_\alpha}))$. At this moment we are not able to show an upper bound but we conjecture that it is ≤ 8. We have generated some examples and all are satisfactory and below of this limit.
Definition 24. *Let $S_{\alpha,\beta} = \{s_1, s_2, ..., s_n\}$ be a sampling of a solid region R. We say that $S_{\alpha,\beta}$ is a interpolating (α, β)-PDS if exists $A \subset S_{\alpha,\beta}$ such that $A \subset \partial R$ and $\partial R \subset \cup_{s_i \in A} B_\alpha(s_i)$. Lets denote a interpolating (α, β)-PDS as $\tilde{S}_{\alpha,\beta}$.*

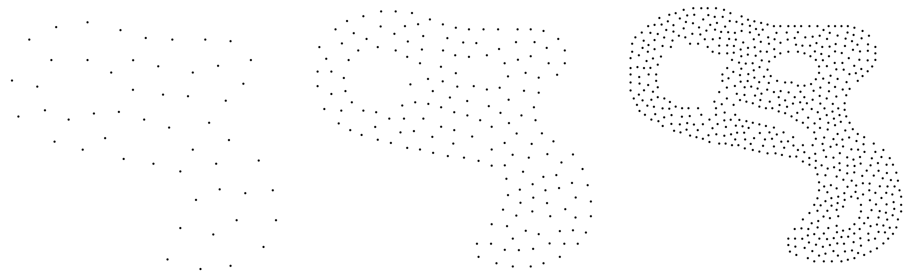
It is easy to see that the (α, β)-PDS from proposition 9 is an interpolationg one according to definition above.

5 Multiresolution with PDS's

After the analysis of one single PDS of a region, from the next section we will start interacting differents PDS's using the scales given by parameter α.

5.1 Scaling Family

For a given region R there is an infinity of PDS's depending on α and on the randomness of the algorithm that generates the sampling. We will introduce the concept of *Scaling Family* which is a particular case of nested family of



(a) (b) (c)

Fig. 6. Exemple of a scaling family with three levels.

points. Such concept has the purpose to structure a set of PDS in such a way that they are nested in order to give us the idea of a sampling in multiresolution with scale change.

Definition 25. *Let $\mathcal{F} = \{S_{\alpha_i}\}_{i\in\{1,2,\ldots n\}}$ be a PDS family of a region R. We say that F is a escaling family if $S_{\alpha_1} \subset S_{\alpha_2} \ldots \subset S_{\alpha_n}$, with $\alpha_1 > \alpha_2 > \ldots > \alpha_n$. For each of the i's we denote de scaling levels and the α_i's of scales.*

For a set of positive reals $\alpha_1 > \alpha_2 > \ldots > \alpha_n$, describe an algorithm from which a scaling family is generated is very simple: Given the sample S_{α_k} it is sufficient to apply the dart throwing algorithm in $(R - \cup_{s_i \in S_{\alpha_k}} B_{\alpha_{k+1}}(s_i))$. In figure 6 we have an example of scaling family with three levels of scale.

An special case of scaling family occurs when the resolution difference between the levels is only of one point.

Definition 26. *Let $\mathcal{F} = \{S_{\alpha_i}\}_{i\in\{1,2,\ldots n\}}$ be a scaling family of R. We say that F is graded if $S_{\alpha_{i+1}} = S_{\alpha_i} \cup \{s_i\}$ with $\alpha_{i+1} < \alpha_i$, $\forall i$.*

If possible, as of a scaling family, to generate one graded family, then we say that such family is *gradable*.

A question that arise from the definition above is: Is every scaling family gradeble? The answer is a negative one and we have a counter example for that in figure 7. In 7.a we have a level of a PDS with radius $\frac{\sqrt{2}}{2} < \alpha < 1$ of a square of side 1. In 7.b we have a level in which the points are over a spacing grid $\frac{1}{3}$ with perturbation. We highlighted the best candidate to the gradation. Under 7.c we exhibited, geometrically, the impossibility of satisfying the coverage condition.

In spite of the fact it is not possible to grade an scaling family, we state that it is always possible for a weaker definition of PDS's that are the (α, β)-PDS's. It is what we will see under section 6.

If there are no sufficient conditions to grade a scaling family, also, there are not assurances to generate a cadenced family using the same algorithm that generates a scaling family, given $\alpha_1 > \alpha_2 > \ldots > \alpha_n$ positive reals. This takes

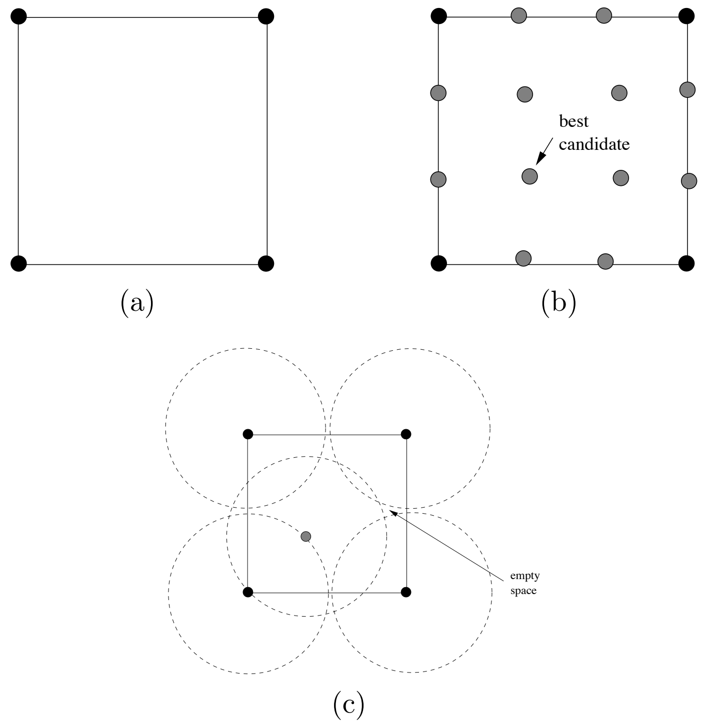


Fig. 7. Counter example for grading process.

place because the scales are already determined, so, as there is the restriction to generate only one sample between two levels, the coverage condition may not be satisfied. In a certain manner, there are jumps or discontinuities between the consecutive levels of the family given by the scales. The good news are that, given a solid object, we are able to generate a graded family with no pre-determined scales. For a better understanding of such graded family generator, lets define a graded family with continuous scales.

Definition 27. *Let* $\mathcal{F} = \{S_{\alpha_t}\}_{t\in[a,b]}$ *be a family of PDS's of* R*. we say that* F *is continuously graded if exist* $b = \alpha_1 > \alpha_2 > ... > \alpha_n = a$ *such that* $S_{\alpha_{i+1}} = S_{\alpha_i} \cup \{s_i\}$ *and if* $s \in (\alpha_{i+1}, \alpha_i)$ *then* $S_s = S_{\alpha_i}$*.*

It is more natural, from the definition above, to think of an algorithm that generates a family continuously graded, given $b > a > 0$. For such, we will start with one PDS of radius $b > 0$. Now define $f : [a, b] \to \mathbb{R}$ such that $f(t) = vol(R - \mathcal{B}_t)$, with $\mathcal{B}_t = \cup_{s_i \in S_{\alpha_t}} B_{\alpha_t}(s_i)$. Observe that in the beginning we have $f(b) = 0$ until $t = t_0$ when then the function f will assume increasing values. At this time t_0 an infinitesimal hole was opened and the coverage condition was undone for disks of radius t_0. However, when we insert a new point p in this hole we will have both Poisson and covering conditions satisfied for the same radius t_0. We repeat the same process until we reach the lowest possible radius a.

5.2 Scaling Family of Meshes

The scaling family of meshes we will introduce as follows have properties that are directly associated to a scaling family. They are a particular case of a nested family of meshes.

Definition 28. *Let S_α be a PDS of a region R and K_α a solid siplicial complex. If $K_\alpha^0 \subset S_\alpha$ we place (S_α, K_α)(read by simplicity "α-pair").*

Definition 29. *Let $\mathcal{F} = \{S_{\alpha_i}\}_{i\in\{1,2,\ldots n\}}$ be a scaling family of a region R. We say that $\mathcal{M}(\mathcal{F}) = \{(S_{\alpha_i}, K_{\alpha_i})\}_{i\in\{1,2,\ldots n\}}$ is a scaling family of meshes if $(S_{\alpha_i}, K_{\alpha_i})$. Similarly to scaling families, for each i's we denote scale level and their respective α_i's of scales.*

Example 1. *Let $F = \{S_{\alpha_i}\}_{i\in\{1,2,\ldots n\}}$ be a scaling family. We say that $C_\alpha\mathcal{F} = \{(S_{\alpha_i}, \overline{C_{\alpha_i}(S_{\alpha_i})})\}_{i\in\{1,2,\ldots n\}}$ is a family of Scaling Solid Alpha Complexes.*

In this example we have a much stronger structure: besides vertices in common, through properties P1, P2 and P3 of section 2.3, there is also a subset of coincident faces between levels. We will exploit this example on a later opportunity when we present filtering by topological operators.

5.3 Filtering by Topological Operators

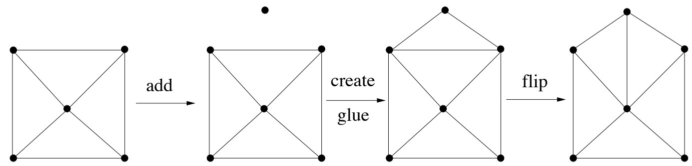
In the PDS's context, to insert points in a mesh means to add more details. If the point is boundary, we will have a better definition of the frontier and if it is interior, we will have a change in the combinatorial structure with an increase in the quantity of triangles. Analogously we also can remove one point of a mesh that consists in lose details. If the point is boundary, we will have a lower definition of frontier and if is interior, there will be a decrease in the quantity of triangles. As it was seen before, the insertion and the removal of points are equivalent, respectively to topological junction and disjunction. In figure 8 we exhibit examples of insertion and removal of points by means of a sequence of topological operators.

Having in hands operations of topological junction or disjunction of points we are able to define a sequence of graded $(S_{\alpha_i}, K_{\alpha_i})$ α-pairs.

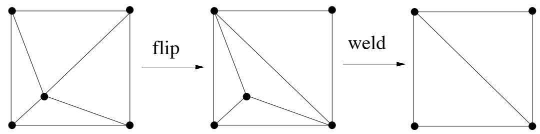
Definition 30. *Filtering by Topological Operators of a region R is a family $\mathcal{M}(\mathcal{F}) = \{(S_{\alpha_i}, K_{\alpha_i})\}_{i\in\{1,2,\ldots n\}}$ of scaling meshes of R such that $\{S_{\alpha_i}\}$ is graded and $(S_{\alpha_{i+1}}, K_{\alpha_{i+1}}) = (S_{\alpha_i}, K_{\alpha_i}) \oplus s_i$ with $s_i = S_{\alpha_{i+1}} - S_{\alpha_i}$.*

Definition 31. *Algorithm by refinement (simplification) of a filtering by topological operators is the one that generates topological operations through a topological junction (disjunction) of points.*

Theorem 1. *For all family of scaling and graded Solids Alpha Complexes $C_\alpha\mathcal{F} = \{(S_{\alpha_i}, C_{\alpha_i}(S_{\alpha_i}))\}_{i\in\{1,2,\ldots n\}}$ there is an algorithm by refining and an*



(a) Example of topological junction.



(b) Example of topological disjuntion.

Fig. 8. Examples of insertion and removal of points.

algorithm by simplification that generate filterings by topological operators of $C_\alpha\mathcal{F}$ *such that refining topological operations are inverse to the simplification topological operations.*

Proof. We will exhibit the two algorithms and later we will state that the sequences generated are inverse.

For each algorithm we will use some auxiliary data structures. In $DT(S_\alpha)$ the faces have a Boolean attribute f that will indicate if they belong or not to the current solid alpha complex. We have a list of operators L_f that store the output of the filtering.

Refinement algorithm: The key idea is exploit the version of Delaunay triangulation construction algorithm which is based on insertion of points (21). We will use a priority queue of half-edges L_{he} that keeps the order in which they will be flipped. The order of the priority queue is given by the power (with the signal inverted) of the new point to be inserted s_i with the circunscribed circle of $he.mate.face.f$, fol all $he \in L_{he}$. In algorithm 1 we have the pseudo code that summarizes all steps described for the insertion of a point.

If during the construction of L_{he} some half-edge he has a non negative key, then it will not be included in L_{he}. Instead of that we will compare the radius of the circunscribed circle of $he.face$ α_i. In case of it is higher and $he.face.f = true$ then we generate the operation destroy and also update $he.face.f$ to $false$. In case of it is lower and $he.face.f = false$ then we generate the operation create and also update $he.face.f$ to $true$. We call this test test_triangle_ref.

In each insertion of point s_i first we test if it falls outside of $Del(S_{\alpha_{i-1}})$ or inside of some face of $DT(S_{\alpha_{i-1}})$. In case it lies inside some face σ such that $\sigma.f = true$ then we generate a split (II.R). The three new triangles will inherit the boolean attribute $f = true$. Before inserting three *half-edges* he_1,

Algorithm 1: Refinement algorithm: inserting a point

Input : s_i, α_i, $DT(S_{\alpha_{i-1}})$

if $s_i \in DT(S_{\alpha_{i-1}})$ **then**
  split(σ, s_i) ;
  for *each new face* F **do**
    $F.f = \sigma.f$;
  if $\sigma.f$ **then**
    generate split(σ, s_i) ;
  for $he = he_1, he_2, he_3 \in \sigma$ **do**
    \ ∗ the function apply_power is described in algorithm 2 ∗ \
    apply_power$(s_i, he.next)$;
else
  for *each he visible to* s_i *in* $\partial DT(S_{\alpha_{i-1}})$ **do**
    $f =$ create$(s_i, he.org, he.dst)$;
    $f.f = falso$;
    apply_power(s_i, he) ;
while $L_{he} \neq \emptyset$ **do**
  $he \longleftarrow L_{he}$.top ;
  $key \longleftarrow he.key$;
  $L_{he}.pop$;
  if $key > 0$ **then**
    if $he.mate.face.f$ *e* $he.face.f$ **then**
      gerar flip(he) ;
    flip(he) ;
    apply_power$(s_i, he.next)$;
    apply_power$(s_i, he.mate.prev)$;
  else
    test_triangle_ref$(he.face)$;
for *cada* $\sigma \in DT(S_{\alpha_i})$ **do**
  test_triangle_ref(σ);

he_2, he_3 in L_{he} we perform a compatibility test following the order given by the key of these half-edges in the priority queue L_{he}. This tests consists in analyzing the boolean attributes f of adjacent faces in a flipping candidate half-edge and follow some rules. The idea is that when both attributes are true, then we generate a flip, which does not take place when both are false. In case they have opposite attibutes then we perform this compatibility. If $he.mate.face.f = true$ then we generate the operator create for $he.face$ and also update $he.face.f$ to $true$. Analogously, if $he.mate.face.f = true$ then we generate the operator destroy to $he.face$ and also update $he.face.f$ to $false$.

To this compatibilization function we name make_compatibility_ref (V.R). See figure 9 and algorithm 3. To each flip there are always two new half-edges to be inserted in L_{he} for which we perform the compatibilization following the counterclockwise order. It is important to emphasize that this test is performed always before the insertion of new half-edges in the queue. If any is not inserted then we perform the test_triangle_ref in $he.face$.

Algorithm 2: function *apply_power*

Input : s_i, half-edge he

```
if not he.edge.is_bdry then
    key ⟵ power(s_i, he.mate.face) ;
    if key > 0 then
        make_compatibility(he) ;
        L_he.insert(he, key) ;
    else
        test_triangle(he) ;
else
    test_triangle(he) ;
```

Algorithm 3: make_compatibility_ref

Input : half-edge he

```
if he.mate.face = true e he.face = false then
    gerar create(he.face) ;
    he.face ⟵ true ;
if he.mate.face = false e he.face = true then
    gerar destroy(he.mate.face) ;
    he.mate.face ⟵ false ;
```

In case point s_i falls inside some face $\sigma \in DT(S_{\alpha_{i-1}})$ such that $\sigma.f = false$ then we do not generate a split operator. Instead of that we generate add (III.R). OThe algorithm proceeds then in the same way, performing flip operations and compatibility tests

In case point s_i lies outside $DT(S_{\alpha_{i-1}})$ then an add operation is generated (IV.R). New faces are inserted with attribute f $false$ by means of create operations following the counterclockwise order of the edges visible from point s_i (see loop of line 11 of algorithm 1). At the same time, operations of compatibility are performed and the priority queue L_{he} is updated. The algorithm then proceeds performing flip operations with compatibility tests.

At last, when the possible operations in the neighborhood of s_i are finished, we sweep the other faces of $DT(S_{\alpha_i})$ and perform the test_triangle_ref in each one of them. The calling order is given by the length of the radius of the

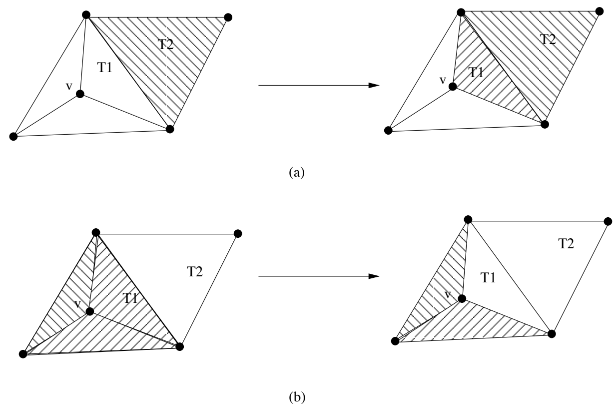


Fig. 9. Cases of the function make_compatibility_ref. v v is the point inserted by a split operation split, T1 and T2 are triangles to be compatibilized. The hachured triangles have a true Boolean attribute and the non hachured have false Boolean attributes. In (a) and (b) we have two possible cases treated by the procedure.

circumscribed circles of each face, starting from the smaller to the longer (I.R)(see loop of line 27, algorithm 1).

Simplification algorithm: The key idea is to use an algorithm to remove the points (18) and adapt it to what we intend. The removal algorithm must take out all triangles incident to s_i and retriangulate in the "Delaunay sense" the star-shaped polygon $H = \{q_0, q_1, ..., q_{k-1}, q_k = q_0\}$ created by these removals. Three consecutive vertices $q_i q_{i+1} q_{i+2}$ along the boundary of H are said to form an *ear* if the segment $q_i q_{i+2}$ lies in H. An ear of H is said of Delaunay if its circumscribed circle does not contain any vertex of H that lies in its interior. The algorithm has the following lemma:

Lemma 2. *Consider polygon $H = \{q_0, q_1, ..., q_k = q_0\}$ and a point p such that the edges $q_i q_{i+1}$ lies in the Delaunay triangulation of $\{q_0, q_1, ..., q_{k-1}, p\}$. If $|power(p, circ(q_i, q_{i+1}, q_{i+2}))|$ is minimal, then $q_i q_{i+2}$ is a Delaunay edge of $\{q_0, q_1, ..., q_{k-1}\}$.*

Proof. See (18). □

As in the refinement algorithm, we have a priority queue L_{ear} of *ears* such that their elements have an augmented structute of type *candidate_ear* composed by three *half-edges*: $s_i q_i$, $s_i q_{i+1}$, $s_i q_{i+2}$ (see below the definition of the structure that will be used in the algorithms 4 and 5). Then L_{ear} keeps the order that the edges $s_i q_{i+1}$ will be flipped, given by the key of the priority queue evaluated as the power of s_i (point to be removed) with circunscribed circle of $q_i q_{i+1} q_{i+2}$. The lemma above assures that the top of the priority queue will contain an ear that belongs to the Delaunay triangulation.

Algorithm 4: Simplification Algorithm: removing a point

Input : s_{i+1}, α_i, $DT(S_{\alpha_{i+1}})$

1 **for** *each* $\sigma \in DT(S_{\alpha_{i+1}} - \mathsf{link}(s_{i+1}))$ **do**
2   test_triangle_spl(σ);
3 generate priority queue L_{ear} ;
4 **while** $(L_{ear}.size > 3$ *and* $\sigma_i \in DT(S_{\alpha_{i+1}}))$ *or* $(L_{ear} \neq \emptyset$ *and* $\sigma_i \in \partial DT(S_{\alpha_{i+1}}))$ **do**
5     $candidate\_ear \longleftarrow L_{ear}.top$;
6     $L_{ear}.pop$;
7   **if** *counter_clockwise*($candidate\_ear$) **then**
8   make_compatibility_spl($candidate\_ear.he_2$) ;
9   **if** $candidate\_ear.he_2.face.f$ *e* $candidate\_ear.he_2.mate.face.f$ **then**
10   generate flip($candidate\_ear.he_2$) ;
11   flip($candidate\_ear.he_2$) ;
12     $L_{ear}.update$;
13 **if** $\sigma_i \in int(DT(S_{\alpha_{i+1}}))$ **then**
14     $test \longleftarrow$ test_triangle_spl(link(s_{i+1})) ;
15   **for** *each neighbor face* F *of* s_{i+1} **do**
16   **if** *test and* **not** $F.f$ **then**
17   generate create($F.f$) ;
18   **if** **not** *test and* $F.f$ **then**
19   generate destroy($F.f$) ;
20   **if** *test* **then**
21   generate weld(σ_i) ;
22   weld(σ_i) ;
23 **else**
24   **for** *each neighbor face* F *of* σ_i **do**
25   **if** $F.f$ **then**
26   generate destroy(F.f) ;
27   destroy(F.f) ;

```
struct Candidate_Ear {
  Half_Edge* he 1, he 2, he 3 ;
}
```

First, before removing a point, we apply the test_triangle_spl in each of the faces of $DT(S_{\alpha_{i+1}})$ outside of the neighbors of s_i by decreasing order of the circumscribed circles (loop of line 1, algorithm 4). In this test we compare the radius of the circumscribed circle of one face σ with α_i. In case it is higher

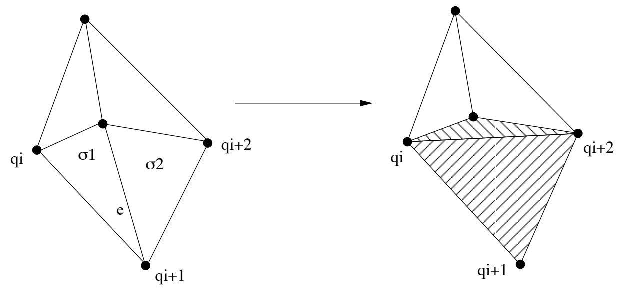


Fig. 10. One of the possible cases of the function make_compatibility_spl. $q_i q_{i+1} q_{i+2}$ is a candidate ear to flip that will be in the Delaunay triangulation. σ_1, σ_2 are faces with false boolean attribute. Operators create are generated para σ_1, σ_2 and after the operator flip flip in the edge e.

and $\sigma.f = true$ we then generate operation destroy and update $\sigma.f$ para $falso$. In case it is lowerand $\sigma.f = false$then we generate the operation create and update $\sigma.f$ to $true$ (I.S). Notice that this test is the inverse equivalent to the refining algorithm procedure test_triangle_ref.

After that, we go applying the operations flip following the order of priority queue L_{ear}. Notice that if $s_i \in DT(S_{\alpha_{i+1}})$ then we will arrive in three ears. Given one ear $q_i q_{i+1} q_{i+2}$ candidate to be fipped, before making it, we perform the test test_triangle_spl in order to verify if it will be in the triangulation. Let σ_1 and σ_2 be the two neighbor faces of $s_i q_{i+1}$ following the counterclockwise order. Depending on the result in the test of $q_i q_{i+1} q_{i+2}$ then we verify the possible compatibilities of σ_1 and σ_2 through a function called make_compatibility_spl (V.S). In other words, this function analyses the faces σ_1 and σ_2 and make them compatible with the result in test of $q_i q_{i+1} q_{i+2}$. If $\sigma_1.f = false$ then we generate the operator create and update $\sigma_1.f$ to $true$. In the same way we make to σ_2. After, we generate a flip. If $q_i q_{i+1} q_{i+2}$ not pass for the test test_triangle_spl then we verify if $\sigma_1.f = true$, we generate the operation destroy and update $\sigma_1.f$ to $false$. The same thing is also made to σ_2. In this case operator flip is not generated. See figure 10.

When all possible flips are performed, if the vertice s_i lies in the interior of $DT(S_{\alpha_{i+1}})$ then it will have valence three. Let q_1, q_2 and q_3 be the three neighbor vertices of s_i. If the triangle $q_1 q_2 q_3$ pass in the test test_triangle_spl then the neighbor faces to s_i that have boolean attribute f as $false$ will generate operators create and will have their attributes f updated to $true$. The orde to generate such operators is given by the power of s_i with adjacen faces to triangel $q_1 q_2 q_3$.After that, we generate the operator weld (II.S). If the triangle $q_1 q_2 q_3$ not pass in the test then we perform the same steps, except generate weld. In this case, we generate the operation destroy to the neighbors triangles of s_i with attribute f as $true$ and update then to $false$. Also they follow the order of the power and after we generate the operator remove(III.S). The same thing happens when the point s_i lies in the boundary. In this case the operations destroy follow the clockwise order(IV.S) (*loop* of line 24, of algorithm 5).

Algorithm 5: Function make_compatibility_spl

```
Input  : ear* candidate_ear
he ⟵ candidate_ear.he_2 ;
if test_triangle_spl(candidate_ear) then
    if he.face.f = falso then
        gerar create(he.face) ;
        he.face.f = verdadeiro ;
    if he.mate.face = falso then
        gerar create(he.mate.face) ;
        he.mate.face.f = verdadeiro ;
else
    if he.face.f = falso then
        gerar destroy(he.face) ;
        he.face.f = falso ;
    if he.mate.face = falso then
        gerar create(he.mate.face) ;
        he.mate.face.f = falso ;
```

Inversion: We identified inverse operations in determined parts of the refining and simplification algorithm. Observe that the flips are ordered in the queues L_{he} and L_{ear} in a symmetrical way, by the point power. There is a symmetry between the functions of compatibilization identified by (V.R), (V.S). It is easy to see also that there is a symmetry in (I.R = I.S^{-1}), (II.R = II.S^{-1}), (III.R = III.S^{-1}) e (IV.R = IV.S^{-1}). □

The two algorithms above may be easily extended for non decreasing monotonous functions $g : \mathbb{R} \to \mathbb{R}$ on the radius of the samplings amostragens in the escaling family. For it, All we need to do is to adapt the test_triangle_ref and test_triangle_spl procedures to compare the radius of the circumscribed circles of the faces segundo according to $g(\alpha)$. This allows us to make a **topological control in the filtering**, i.e., the function g controls all topological changes between the filtering levels.

Corollary 1. *Let $g : \mathbb{R} \to \mathbb{R}$ be a non decreasing monotonous function. Let $\mathcal{F} = \{S_{\alpha_i}\}$ be a graded family and $C_\alpha F = \{C_{g(\alpha_i)}(S_{\alpha_i})\}$ the Solid Alfa Complexes Family of F with function g. Then, there exists a algorithm by refinement and an algorithm by simplification that generate a filtering by topological operetorsof $C_{g(\alpha)}\mathcal{F}$ such that the refinement topological operators are inverse to the simplification topological operators.*

Notice that the Delaunay triangulation is equivalent to the corolary above considering the $g(x) = \infty$.

5.4 Quasi-Scaling Family

So far we have approached PDS's and solid alpha complexes. In this section we will define a more general type of sampling, the (α, β)-ADP's. We will also extend the definition of Solid Alpha Complex to Quasi Solid Alpha Complex where in the latter the topological information of the points (boundary or interior) are taken into consideration. Our main goal is to arrive at Theorem 3 that extends Theorem 1 using these two new extensions.

In the same way we defined the scaling families for PDS's, we can analogously define a Quasi-Scaling Family (QSF) and its derivations.

Definition 32. *Let $\mathcal{F} = \{S_{\alpha_i \beta_i}\}_{i \in \{1,2,...n\}}$ be a family of PDS's of a region R. We say that F is scaling if $S_{\alpha_1 \beta_1} \subset S_{\alpha_2 \beta_2} ... \subset S_{\alpha_n \beta_n}$, such that $\beta_1 > \beta_2 > ... > \beta_n$. For each one of i's we denote as scaling levels and itss β_i's as scales.*

Below we present a scheme of an algorithm that builds a particular Quasi-scaling family.

Proposition 10. *Given $\alpha_1 > 0$, we can generate a quasi-scaling famliy $\mathcal{F} = \{S_{\alpha_i \beta_i}\}_{i \in \{1,2,...n\}}$ of a region R such that:*

(1) $\alpha_{i+1} = \alpha_i / 2$;
(2) $\beta_{i+1} = \beta_i / 2$;
(3) $\beta_i = \alpha_i / 2$.

with the following steps:

(1) Do $i = 1$; use proposition 9 to generate $S_{\alpha_i \beta_i}$; do $i = i + 1$ and $\alpha_i = \alpha_i / 2$;
(2) Use proposition 9 to generate $S_{\alpha_i \beta_i} - S_{\alpha_{i-1} \beta_{i-1}} \subset (R - \cup_{s_j \in S_{\alpha_{i-1} \beta_{i-1}}} B_{\beta_i}(s_j))$;
(3) Do $\alpha_{i+1} = \alpha_i / 2$; $i = i + 1$, and go back step 1 until $i = n$;

Proof. According to proposition 9 it follows that $\beta_i = \alpha_i / 2$ for each i. □

5.5 Grading a Quasi-Scaling Family

We exposed the difficulties of grading a scaling family. To our surprise, due to the greater generality of a quasi-scaling family this is always possible. The main result on grading a QEF is theorem 2. First place we will define when a QEF is graded.

Definition 33. *Let $F = \{S_{\alpha_i \beta_i}\}_{i \in \{1,2,...n\}}$ be a Quasi-Scaling Family of R. We say that F is Quasi-Graded if $S_{\alpha_{i+1} \beta_{i+1}} = S_{\alpha_i \beta_i} \cup \{s_i\}$, $\beta_{i+1} < \beta_i$, $\alpha_{i+1} < \alpha_i$*

with $\beta_{i+1} = d(S_{\alpha_i\beta_i}, s_i)$.

Theorem 2. *Let* $\mathcal{F} = \{S_{\alpha_i\beta_i}\}_{i\in\{1,2,...n\}}$ *be a Quasi-scaling family of R. Then exists a family* $\mathcal{F}_c$ *that is a grading* $\mathcal{F}$.

Proof. We will show two ways of grading a Quasi-Scaling Family. In the first one we order directly by inserting points. In the second one we order the by removing points.

1^a) Let $D = S_{\alpha_{i+1}\beta_{i+1}} - S_{\alpha_i\beta_i} = \{s_1, s_2, ..., s_k\}$, then we generate a ordering $D_\sigma = \{s_{\sigma(1)}, s_{\sigma(2)}, ..., s_{\sigma(k)}\}$ such that by the unicity of the distance $\exists!\, s_{\sigma(l)} \in D$ hold:

$$\max\{d(s, A_{l-1}) \quad | \quad s \in D_{l-1}\} = \beta_{i_l}$$

where, $D_l = D - \{s_{\sigma(1)}, s_{\sigma(2)}, .., s_{\sigma(l)}\}$ e $A_l = S_{\alpha_i\beta_i} \cup \{s_{\sigma(1)}, s_{\sigma(2)}, .., s_{\sigma(l)}\}$.

Here we will make two observations. In the first one we consider$DT(A_l)$.

The value of α_{i_l} can be chosen as the lowest real positive that satisfies the covering condition:

$$\alpha_{i_l} = \inf\{\alpha \in \mathbb{R} \,|\, R \subset \cup_{s\in A_l} B_\alpha(s)\}.$$

2^a) Given $S_{\alpha_i,\beta_i} \cup \{\sigma_1, \sigma_2, ..., \sigma_k\} = S_{\alpha_{i+1},\beta_{i+1}}$, the grading algorithm of $\mathcal{F}$ is inintialized from the Delaunay triangulation of $S_{\alpha_{i+1},\beta_{i+1}}$. We perform the ordering using once again the algorithm to remove the points described in Devillers (18).

We initially built a priority queue L_e that contains all edges of the triangulation such that the top is the smallest one. Let $lv(w)$ be the level of the point w and link(w) the set composed by the neighbor edges to w in the mesh. Let e be the edge at the topo of the priority queue L_e. The point to be removed is one of the vertices of e. Let p and q be such vertices. Then if $lv(p) > lv(q)$ we remove p as described in (18). Analogously, if $lv(q) > lv(p)$ then we remove q. In case of equality between the levels, we decided for removing the one that is nearest to the neighbor vertices, not considering edge e. Our implementation about "neighbor proximity measure" for any vertice is the lenght of the gratest edge in its link. The longer the length, longer is the distance. Therefore, if $lv(p) = lv(q)$ consider e_p and e_q edges such that $e_p = \max\{m(w)|w \in \text{link}(p)\}$ and $e_q = \max\{m(w)|w \in \text{link}(q)\}$, where $m(w)$ is the length of w. If $m(e_p) > m(e_q)$ then we remove q otherwise, removes p.

Algorithm 6: Ordenation by removals

Input : $S_{\alpha_{i+1},\beta_{i+1}}$.
Output: Ordenation of $S_{\alpha_{i+1},\beta_{i+1}} - S_{\alpha_i,\beta_i}$.

```
A ⟵ S_{α_{i+1},β_{i+1}} − S_{α_i,β_i} ;
M ⟵ DT(S_{α_{i+1},β_{i+1}});
Create the a priority queue L_e ;
while A ≠ ∅ do
    e ⟵ topo de L_e ;
    if lv(p) > lv(q) then
        remove p from DT(S_{α_{i+1},β_{i+1}}) ;
        A ⟵ A − {p} ;
    if lv(q) > lv(p) then
        remove q from DT(S_{α_{i+1},β_{i+1}});
        A ⟵ A − {q} ;
    if lv(q) = lv(p) then
        e_p = max{m(w) | w ∈ link(p)} ;
        e_q = max{m(w) | w ∈ link(q)} ;
        if e_p > e_q then
            remove q from DT(S_{α_{i+1},β_{i+1}});
            A ⟵ A − {q} ;
        else
            remove p from DT(S_{α_{i+1},β_{i+1}});
            A ⟵ A − {p} ;
    Update L_e ;
```

The queue L_e is updated and the algorithm proceeds in the same way. Notice that the value of β corresponds to the length of the top edge in L_e. See algorithm 6.

In both cases, we generate "intermediary" sublevels between the levels of a Quasi-Scaling Family to turn it into a Quasi-Graded Family. □

5.6 Quasi-Scaling Family of Meshes

Lets now define some structures associated to QSF's.

Definition 34. *Let $S_{\alpha\beta}$ be a (α, β)-PDS of a region R and $K_{\alpha\beta}$ a solid simplicial complex. We say that $(S_{\alpha\beta}, K_{\alpha\beta})$ is a (α, β)-pair if $K^0_{\alpha\beta} \subset S_{\alpha\beta}$.*

Definition 35. *Let $\mathcal{F} = \{S_{\alpha_i\beta_i}\}_{i\in\{1,2,\dots n\}}$ be a quasi-scaling family of a region R. we say that $\mathcal{M}(\mathcal{F}) = \{(S_{\alpha_i\beta_i}, K_{\alpha_i\beta_i})\}_{i\in\{1,2,\dots n\}}$ is a Quasi-Scaling Family of Meshes if $(S_{\alpha_i\beta_i}, K_{\alpha_i\beta_i})$ is a (α, β)-pair for all i. As in scaling families, for each of i's we denote them as scaling levels and its β_i's as scales.*

Example 2. *Let $F = \{\tilde{S}_{\alpha_i\beta_i}\}_{i\in\{1,2,\dots n\}}$ be a Quasi-Scaling Family. We say* *that* $QC_\alpha\mathcal{F} = \{(\tilde{S}_{\alpha_i\beta_i}, C_{\alpha_i}(\tilde{S}_{\alpha_i\beta_i}))\}_{i\in\{1,2,\ldots n\}}$ *is a Quasi-Scaling Family of Solid Alpha Complexes(QSFSAC) where* $QC_\alpha(\tilde{S}_{\alpha_i\beta_i})$ *is the quasi solid alpha complex of* $\tilde{S}_{\alpha_i\beta_i}$.

Based on proposition 10 we can conclude four important facts about QSFSAC's.

- The Hausdorff distance of the boundary is limited by α_i in each level, assuring the convergence of the approximation of the solid region by QACS';
- The convergence of the sequence given by the QSFSAC is of good approximations and therefore has convergence of normals;
- In each sampling level its respective QSAC have compatibility with the topological information i.e. if $s \in int(R)$ then $s \in int(C_\alpha(R))$.
- in practice, the set of triangles have aspect ratio lower than 8.

5.7 *Returning to Filtering*

Equivalent to Theorem 1 for any QSFSAC not necessarily graded, we have the following theorem:

Theorem 3. *For any non decreasing monotonous function* $g : \mathbb{R} \to \mathbb{R}$ *and given a QSFSAC* $QC_{g(\alpha)}\mathcal{F} = \{(\tilde{S}_{\alpha_i\beta_i}, QC_{\alpha_i}(\tilde{S}_{\alpha_i\beta_i}))\}_{i\in\{1,2,\ldots n\}}$, *exists an algorithm by refinement and an algorithm by simplification that generate topological operators of* $QC_\alpha\mathcal{F}$ *such that the refinement operations are inverse to the simplification operations.*

Proof. First we apply theorem 2 to grade intermediate levels of F in a new family F_c. The algorithm is the same of theorem 1 with the modification of the function test_triangle_ref to refinement and the function test_triangle_spl to simplification. In these functions the validation for one face be included or not are given by the parameter $g(\alpha_i)$ and by 2 of definition 22. □

The function g allows a certain degree of control to the algorithms in the construction of filtering by topological operators. As saw before if the function $g = \infty$ then the filtering corresponds to the Delaunay triangulation.

6 Conclusion

In this article we introduced a theoretical and practical framework for multiresolution with topology change control. As far as we know there are no works on this matter.

We analysed the boundary geometry and its topology through Poisson Disks Sampling (PDS's and (α, β)-PDS's) recovered by $s$olid alpha complexes (and their variants) for each resolution level. With that we were able to insert atomic operations such as stellar and handle between the resolution levels. To this sequence of operations we gave the name of topological operations filtering.

In figure 11 we have from on the right four resolution levels of a solid region shaped as rectangle with two holes that are circles with distinct radius. The levels were sampled according proposition 10. As it was expected, as we increased the resolution, the holes go appearing as well as their respective boundaries go being more detailed. On the left are represented intermediary levels between levels three and four. The points were ordered according to algorithm 6.

The main result of the article brings a theorem (in two versions) that says if it is possible to generate two filterings by topological operators either by simplification or refining in an independent way with the property that the sequence of operations are inverse. This theorem will allow the generation of a representation of one mesh in variable resolution (12) that is flexible so that functions of adaptation may generate adaptative meshes starting both from the more simplified level and from the more refined level, depending on the application. It is what we will do in one of our future works. They are:

- To generate a hierarchical structure between the resolution levels to comprise a mesh in variable resolution;
- To create adaptation functions;
- To generate applications;
- To generalize all results for dimension three.

Acknowledgements

We thank to CNPq for all financial support and to IMPA for the structure and motivation. We thank Dimas Martinez for reviews and suggestions.

References

[1] J. Gomes, B. Costa, L. Darsa e L. Velho. *Graphical Objects*. The Visual Computer, 12(6): 269–282, 1996.

[2] N. Amenta, M. Bern and D. Eppstein. *The Crust and the β-Skeleton: Combinatorial Curve Reconstruction*. Graphical Models and Image Processing, 60/2:2, 125–135, 1998.

[3] F. Bernardini and C. Bajaj. *Sampling and Reconstructing Manifolds Using Alpha-Shapes*. Proc. of the Ninth Canadian Conference on Computational Geometry, 193–198, August, 1997.

[4] F. Bernardini, C. Bajaj, J. Chen and D. Schikore. *Automatic Reconstruction of 3D CAD Models form Digital Scans*. Int. J. on Comp. Geom. and Appl., 9(4–5):327–370, August–October 1999.

[5] C. E. Shannon. *Communication in the presence of noise*. Proc. Institute of Radio Engineers, 37(1): 10–21, 1949.

[6] E. Medeiros, L. Velho and H. Lopes. *Restricted BPA: Applying Ball-Pivoting on the Plane*. Proc. of XVII Simpsio Brasileiro em Computao Grfica e Processamento de Imagens (SIBGRAPI), Curitiba-Brazil, 2004.

[7] E. Medeiros, T. Lewiner, H. Lopes, and L. Velho. *An Aspect Ratio Upper Bound in 2D Solid Alpha Complexes of Poisson Disc Samplings*. Technical Report 02-07, IMPA - VISGRAF Laboratory, 2007.

[8] E. Medeiros, T. Lewiner, H. Lopes, and L. Velho. *Reconstructing Poisson Disc Samplings of Solid Objects with Topological Guarantees*. Technical Report 03-07, IMPA - VISGRAF Laboratory, 2007.

[9] L. Kobbelt. $\sqrt{3}$-*Subivision*. Proc. do SIGGRAPH, 2000.

[10] L. Velho e D. Zorin. *4–8 Subdivision*. Computer-Aided Geometric Design, 18(5):397–427, 2001.

[11] L. Velho, H. Lopes, E. Medeiros, T. Lewiner and G. Tavares. *Topological Mesh Operators*. Technical Report - VISGRAF Laboratory TR-2005-03, IMPA, October 2005.

[12] E. Puppo. *Variable Resolution Triangulations*. Computational Geometry, 11(3–4): 219–238, 1998.

[13] H. Hoppe. *Progressive meshes*. Proc. do SIGGRAPH, 1996.

[14] H. Edelsbrunner, D. Letscher e A. Zomorodian. *Topological Persistence and Simplification*. Discrete Comput. Geom., 28: 511–533, 2002.

[15] H. Edelsbrunner. *The Union of Balls and ts Dual Shape*. Discrete Comput. Geom. 13: 415–44, 1995.

[16] H. Edelsbrunner e E. P. Mucke. *Three-Dimensional Alpha Shapes*. ACM Trans. Graphics 13: 43–72, 1994.

[17] M. de Berg, M. van Kreveld, M. Overmars e O. Schwarzkopf. *Computational Geometry*, Springer-Verlag, Berlin, 1997.

[18] O. Devillers. *On Deletion in Delaunay Triangulation*. Internat. J. Comput. Geom. Appl., 12:193-205, 2002.

[19] R. L. Cook. *Stochastic Sampling in Computer Graphics*. ACM Transactions on Graphics, 5(1):51-72, January, 1986.

[20] S. Hiller and O. Deussen. *Tiled Blue Noise Samples*. Proceedings of VISION, MODELING, AND VISUALIZATION, IOS Press, pp. 265–271, 2001.

[21] C. L. Lawson. *Software for C1 Surface Interpolation*. Mathematical Software III (John R. Rice, editor), Academic Press, pp. 161-194., New York, 1977.

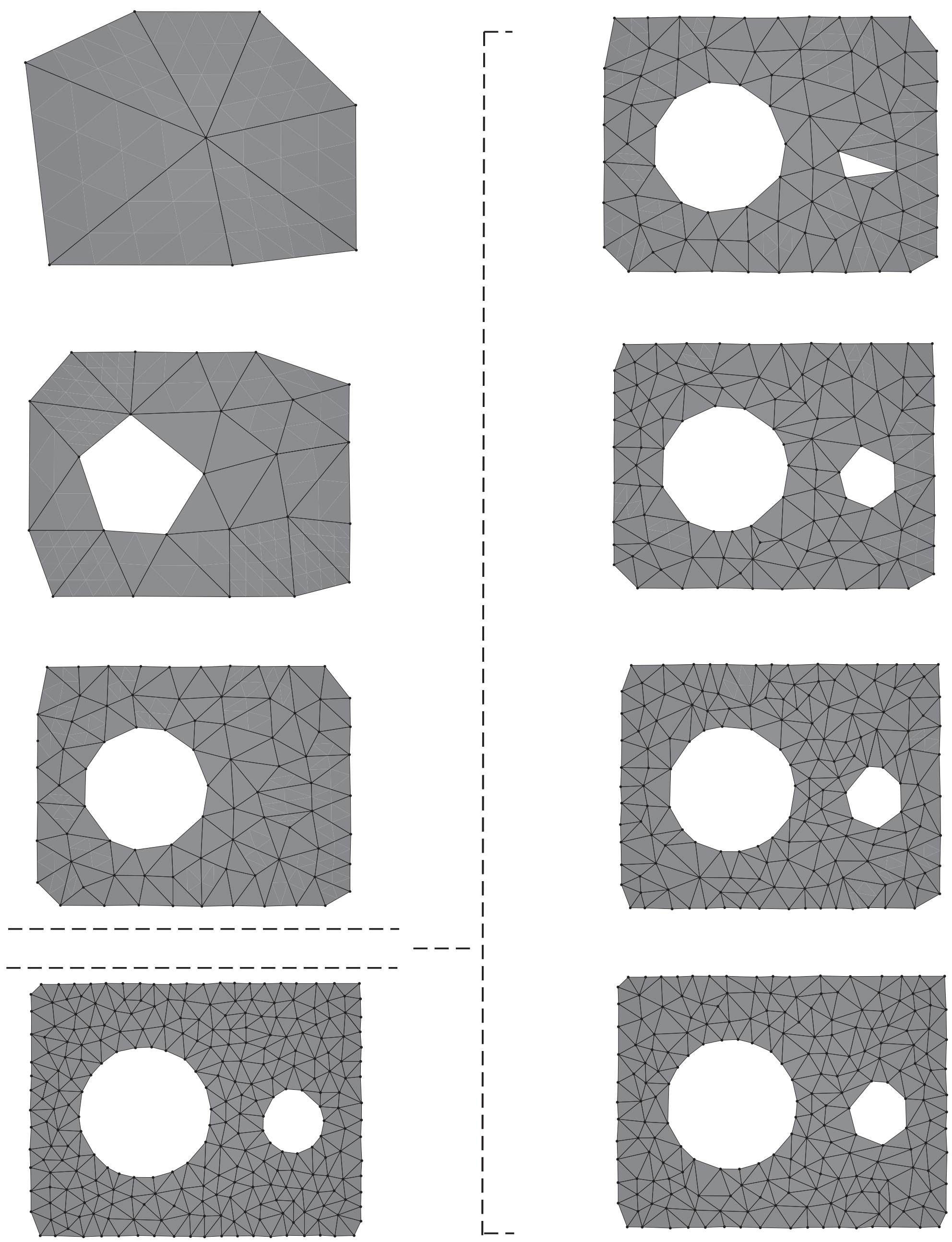


Fig. 11. Four levels (left) and four sublevels (right).