

Superimposing Voronoi Complexes for Shape Deformation

[Extended Abstract]

Chao Chen
School of Computing
National University of Singapore
chencha1@comp.nus.edu.sg

Ho-Lun Cheng
School of Computing
National University of Singapore
hcheng@comp.nus.edu.sg

ABSTRACT

Edelsbrunner et al. defined a framework of shape deformations with shapes bounded by skin manifold. We prove that the infinitely many synthesized shapes in the deformation sequence share finitely many common Voronoi complexes. Therefore, we propose a new algorithm to compute the common Voronoi complexes efficiently for the deformations, and use these common complexes to compute the synthesized shapes in real time. This makes generating, visualizing, and customizing shape deformations feasible.

1. INTRODUCTION

Edelsbrunner et al. defined a framework of shape space construction, in which new shapes are synthesized with a set of reference shapes by mixing them. This allows the deformation of one shape into another [2]. This is a robust way of shape space construction for fields such as molecular and geometric modeling.

The shapes employed by Edelsbrunner et al. are represented as *skin bodies*, each of which is defined by a set of weighted points [4]. Each new instance of the synthesized shape, namely, the *intermediate shape*, is generated by firstly computing its weighted point set. From its Voronoi and Delaunay complexes, the skin body can be constructed. The bottleneck is that for each intermediate shape, its Voronoi complex is computed individually, with time $O(m \lg m + m^{\lceil d/2 \rceil})$, in which m is the cardinality of the weighted point set. Therefore, it is impractical to generate and visualize shape deformation, which is approximated by a sequence of intermediate shapes synthesized with the initial and final shapes.

In this paper, we improve the efficiency of computing the intermediate shapes and make shape deformation feasible. We prove that in \mathbb{R}^d space, all the infinite number of intermediate shapes share finitely many Voronoi complexes. Basing on this finding, we design an algorithm to compute the common Voronoi complexes with those of the reference shapes. Our algorithm is faster than computing the Voronoi com-

plex directly when the dimension $d > 2$. Beside the better efficiency of computing the Voronoi complex, most intermediate shapes can be computed immediately by reusing the generated common Voronoi complexes. This is especially helpful in the generation and visualization of shape deformations, which require a lot of intermediate shapes with only a few common Voronoi complexes.

See Figure 1.1 for an example. To generate a deformation of a shape of the letter ‘X’ into a shape of the letter ‘I’, we require a lot of intermediate shapes, each of which is a mixture of the reference shapes, ‘X’ and ‘I’. Without our algorithm, we have to compute the Voronoi complex for each instance of the intermediate shapes. However, with our algorithm, we only generate one common Voronoi complex with those of the reference shapes, ‘X’ and ‘I’. With this common Voronoi complex, any intermediate shape in the deformation is immediate.

Moreover, with our algorithm, we are able to customize the shape deformations by introducing additional reference shapes, which is impossible before because of the unavailability of the visualization. See Figure 1.2. We influence the deformation of the shape of letter ‘X’ into the shape of letter ‘I’ with a shape of letter ‘O’. By applying a certain positive influence of the shape ‘O’, we can make a hole in the center of the deforming shape. On the contrary, if we apply a negative influence of the shape ‘O’, we can make the deforming shape more compact. With our algorithm, all these intermediate shapes share only two common Voronoi complexes, which are computed with the Voronoi complexes of the reference shapes, ‘X’, ‘I’, and ‘O’.

Outline.

In Section 2, we introduce the skin and its construction. In Section 3, we define the the intermediate shape, and present a theorem stating that there are only finitely many Voronoi complexes for all the intermediate shapes. Finally, in Section 4, we design a new algorithm to compute the Voronoi complexes of the intermediate shapes.

2. BACKGROUNDS

This section serves for the purpose of stating the notations for later sections. We introduce the geometric foundation of the skin body. The skin body is bounded by the *skin*, a compact manifold without boundary in any dimension. It is defined on a weighted point set, and constructed on its Voronoi and Delaunay complexes.

We will not go to the details of skin manifold. Interested readers could refer other literatures.

2.1 Complexes

We first introduce the Delaunay and Voronoi complexes of weighted points.

Weighted points.

A *weighted point* in \mathbb{R}^d can be written as $b_i = (z_i, w_i)$, where $z_i \in \mathbb{R}^d$ is its position and $w_i \in \mathbb{R}$ is its weight. We can also view a weighted point, b_i , as a ball in \mathbb{R}^d with center z_i and radius $\sqrt{w_i}$. For a set of balls, $B_0 \subset \mathbb{R}^d \times \mathbb{R}$, we define the *union of balls* as

$$\bigcup B_0 = \bigcup_{b_i \in B_0} \{x \in \mathbb{R}^d \mid \|xz_i\|^2 \leq w_i\}.$$

Voronoi and Delaunay complexes.

The *weighted distance* of a point $x \in \mathbb{R}^d$ from a weighted point, b_i , is defined as

$$\pi_{b_i}(x) = \|xz_i\|^2 - w_i. \quad (1)$$

Given a finite set B_0 , the *Voronoi region* for each weighted point, $b_i \in B_0$, is

$$\nu_i = \{x \in \mathbb{R}^d \mid \pi_{b_i}(x) \leq \pi_{b_j}(x), \forall b_j \in B_0\}.$$

We define the non-empty intersection of m Voronoi regions as the *Voronoi cell* of a set of weighted points $X \subseteq B_0$, namely,

$$\nu_X = \bigcap_{b_i \in X} \nu_i.$$

The collection of all the non-empty Voronoi cells is called the *Voronoi complex* of B_0 ,

$$V_{B_0} = \{\nu_X \mid X \subseteq B_0, \nu_X \neq \emptyset\}.$$

For any $B_0 \subseteq \mathbb{R}^d \times \mathbb{R}$, $z(B_0)$ is the set of centers of B_0 . For each $\nu_X \in V_{B_0}$, we define its corresponding *Delaunay cell*, δ_X , as the convex hull of the set of centers of X , $\text{conv}(z(X))$. The collection of all the Delaunay cells is called the *Delaunay complex* of B_0 , namely,

$$D_{B_0} = \{\delta_X \mid \nu_X \in V_{B_0}\}.$$

Figure 1(a) shows an example of the Voronoi and Delaunay complexes of a weighted point set. Generally, in \mathbb{R}^2 , the Delaunay cells are all simplices, namely, vertices, edges and triangles.

Furthest-neighbor Voronoi complex.

Generally, the Voronoi complex can be defined on different distance functions. If we use $-\pi_{b_i}(x)$ as the distance function, the resulting Voronoi complex is the *furthest-neighbor Voronoi complex*. Each of its elements, a *furthest-neighbor Voronoi cell*, contains points with the longest weighted distances from the weighted points. Hence, we can define the *furthest-neighbor Delaunay complex* and *furthest-neighbor Delaunay cell*.

The furthest-neighbor Voronoi complex will be used to define the Voronoi complex of the intermediate shape in the next section.

2.2 Skin

In this subsection, we let $B_0 = \{b_i \mid i = 1..n_0\} \subset \mathbb{R}^d \times \mathbb{R}$, and give the definition of the skin of B_0 , as well as its construction.

Before defining the skin, three operations on weighted points (or balls) are given. The aim is to establish the operations for linear combinations of weighted points [4]. For $b_i, b_j \in B_0$ and $\gamma \in \mathbb{R}$, the *addition*, *scalar multiplication* and *square root* of weighted points are defined as

$$\begin{aligned} b_i + b_j &= (z_i + z_j, w_i + w_j + 2\langle z_i, z_j \rangle), \\ \gamma b_i &= (\gamma z_i, \gamma w_i + (\gamma^2 - \gamma)\|z_i\|^2), \quad \text{and} \\ \sqrt{b_i} &= (z_i, w_i/2), \end{aligned}$$

respectively, in which $\langle z_i, z_j \rangle$ is the scalar product of two vectors z_i and z_j . For $\gamma_i \in \mathbb{R}$, the *linear combination* of B_0 is $\sum_{i=1}^{n_0} \gamma_i b_i$. It is a *convex combination* if $\sum_{i=1}^n \gamma_i = 1$ and $\gamma_i \geq 0$. The *convex hull* of B_0 , $\text{conv}(B_0)$, is the set of all the convex combinations of the elements of B_0 .

By extending the *square root* to a set of weighted points, we define the *skin* in \mathbb{R}^d as

$$\text{skin}(B_0) = \partial(\bigcup \sqrt{\text{conv}(B_0)}).$$

The skin of B_0 is homeomorphic to the boundary of $\bigcup B_0$ [4]. We define the shape bounded by skin as the *skin body*, namely, $\text{body}(B_0) = \bigcup \sqrt{\text{conv}(B_0)}$. Figure 1(c) shows an example of the skin body in \mathbb{R}^3 .

The skin manifold could be computed immediately if the Voronoi complex is determined. See Figure 1(b) for a skin computed on a given Voronoi complex.

3. INTERMEDIATE SHAPES AND THEIR VORONOI COMPLEXES

In this section, we start with the definition of the intermediate shapes. After that, we present a theorem about the Voronoi complexes of the intermediate shapes. This theorem states that all the intermediate shapes share finitely many Voronoi complex.

3.1 Intermediate Shapes

Given a finite set of given shapes, namely, the *reference shapes*, an infinite family of intermediate shapes can be constructed. Each intermediate shape is the mixture of the reference shapes, with a set of weights signaling the similarities between the mixture and the reference shapes.

Let the number of reference shapes be n . Let $\mathfrak{B} = \{B_1, \dots, B_n\}$ be the set of weighted point sets defining the set of reference shapes, $S = \{\text{body}(B_i) \mid B_i \in \mathfrak{B}\}$. To define the intermediate shapes of S , we firstly define the affine combinations of \mathfrak{B} , namely, $B(c)$.

Given two weighted point sets, B_0 and B_1 , we define the *addition*, and *scalar multiplication* of them as

$$\begin{aligned} B_0 + B_1 &= \{b_i + b_j \mid b_i \in B_0, b_j \in B_1\}, \quad \text{and} \\ \gamma B_0 &= \{\gamma b_i \mid b_i \in B_0\}, \quad \text{for } \gamma \in \mathbb{R}, \end{aligned}$$

respectively. With these operations, we can define the *linear combination* of \mathfrak{B} as $B(c) = \sum_{i=1}^n \gamma_i B_i$ with the *coefficient*

vector, $c = (\gamma_1, \dots, \gamma_n) \in \mathbb{R}^n$. If $\sum_{i=1}^n \gamma_i = 1$, $B(c)$ is an affine combination of \mathfrak{B} , namely, the *intermediate weighted point set*. Note that the cardinality of $B(c)$ is no more than $\prod_{i=1}^n \text{card}(B_i)$.

An *intermediate shape* is defined as the skin shape bounded by the skin of $B(c)$, namely, $\text{body}(B(c))$. Each intermediate shape corresponds to a coefficient vector, c , which states the weights of all these reference shapes in this mixture. All the coefficient vectors of the intermediate shapes form a hyperplane $\Gamma \subseteq \mathbb{R}^n$. A *deformation* of one intermediate shape into another is parameterized with a path connecting the two corresponding coefficient vectors in Γ . Note that each reference shape is also an intermediate shape, whose corresponding coefficient vector is an unit basis vector in \mathbb{R}^n .

3.2 Intermediate Voronoi Complexes

We denote the Voronoi complex of $B(c)$ as $V(c)$, namely, the *intermediate Voronoi complex*. To compute an intermediate shape, $\text{body}(B(c))$, the corresponding intermediate Voronoi complex, $V(c)$, is required. However, it is computationally expensive if we compute $V(c)$ for each individual value of c , especially in deformation of the intermediate shapes. On the other hand, we will show that Γ can be divided into finite number of partitions, and in each partition, the intermediate Voronoi complexes are the same for all the possible values of c .

For convenience, we assign signs to the Voronoi complex and its elements. For a weighted point set, B_0 , its Voronoi complex with '+' sign, $V_{B_0}^+$, is the nearest-neighbor Voronoi complex of B_0 , and $V_{B_0}^-$ is the furthest-neighbor Voronoi complex of B_0 . The same rule applies to the Voronoi cells ν_X^+ and ν_X^- .

Each weighted point $b(c) \in B(c)$ is an affine combination of n weighted points from the n given $B_i \in \mathfrak{B}$, namely, $b(c) = \sum_{i=1}^n \gamma_i b_i$, for some $b_i \in B_i$. In order to define the Voronoi region of $b(c)$ with respect to $B(c)$, we firstly derive the weighted distance of any point $x \in \mathbb{R}^d$ from $b(c)$.

L. e

t $b(c) = \sum_{i=1}^n \gamma_i b_i$, such that $\sum_{i=1}^n \gamma_i = 1$. The weighted distance of any point $x \in \mathbb{R}^d$ from $b(c)$ is the affine combination of the weighted distances of x from b_1, \dots, b_n , namely,

$$\pi_{b(c)}(x) = \sum_{i=1}^n \gamma_i \pi_{b_i}(x).$$

PROOF. We prove this lemma by induction on the number n . For $n = 1$, it is true since $\gamma_1 = 1$ and $B(c) = B_1$. When $n = 2$, $b(c)$, its center and weight are

$$\begin{aligned} b(c) &= \gamma_1 b_1 + (1 - \gamma_1) b_2, \\ z_{b(c)} &= \gamma_1 z_1 + (1 - \gamma_1) z_2, \quad \text{and} \\ w_{b(c)} &= (\gamma_1 w_1 + (\gamma_1^2 - \gamma_1) \|z_1\|^2) + \\ &\quad ((1 - \gamma_1) w_2 + ((1 - \gamma_1)^2 - (1 - \gamma_1)) \|z_2\|^2) + \\ &\quad 2\langle \gamma_1 z_1, (1 - \gamma_1) z_2 \rangle, \end{aligned}$$

respectively. Substitute them into the weighted distance function (1). For any $x \in \mathbb{R}^d$, we have

$$\pi_{b(c)}(x) = \gamma_1 \pi_{b_1}(x) + (1 - \gamma_1) \pi_{b_2}(x). \quad (2)$$

The claim is true for $n = 2$.

Assume the claim is true for $n = k$. When $n = k+1$, without loss of generality, we assume $\gamma_{k+1} \neq 1$. Let $\mathfrak{B}' = \mathfrak{B} - \{B_{k+1}\}$ and

$$c' = \left(\frac{\gamma_1}{1 - \gamma_{k+1}}, \frac{\gamma_2}{1 - \gamma_{k+1}}, \dots, \frac{\gamma_k}{1 - \gamma_{k+1}} \right).$$

Let $B'(c')$ be the affine combination of \mathfrak{B}' , and $b'(c') \in B'(c')$. Then, $b(c)$ can be expressed as the affine combination of $b'(c')$ and b_{k+1} , namely,

$$b(c) = (1 - \gamma_{k+1}) b'(c') + \gamma_{k+1} b_{k+1}.$$

From Equation (2), and $n = k$ assumption, the weighted distance of any point $x \in \mathbb{R}^d$ from $b(c)$ is exactly

$$\begin{aligned} \pi_{b(c)}(x) &= (1 - \gamma_{k+1}) \pi_{b'(c')}(x) + \gamma_{k+1} \pi_{b_{k+1}}(x) \\ &= \sum_{i=1}^{k+1} \gamma_i \pi_{b_i}(x), \end{aligned}$$

as required. \square

We are now ready to give a new method to determine the Voronoi region of $b(c)$ with respect to $B(c)$. We start constructing $V(c)$ from a simple situation, in which $\gamma_i > 0$ for all i .

F. o

r any $b(c) \in B(c)$ with all $\gamma_i > 0$, its Voronoi region with respect to $B(c)$ is the intersection of the Voronoi regions of b_1, \dots, b_n with respect to B_1, \dots, B_n , respectively, namely,

$$\nu_{b(c)} = \bigcap_{i=1}^n \nu_{b_i}. \quad (3)$$

PROOF. It is easy to see that $\bigcap_{i=1}^n \nu_{b_i} \subseteq \nu_{b(c)}$, because $\gamma_i \pi_{b_i}(x) \leq \gamma_i \pi_{b'_i}(x)$ for all $b'_i \in B_i$.

Next we prove $\nu_{b(c)} \subseteq \bigcap_{i=1}^n \nu_{b_i}$. We are going to prove that for any $k = 1..n$, and any point $x \in \mathbb{R}^d$, $x \in \nu_{b(c)}$ implies that $x \in \nu_{b_k}$. For any $b'_k \in B_k$, let

$$b'(c) = \sum_{i=1..n, i \neq k} \gamma_i b_i + \gamma_k b'_k.$$

We have $\pi_{b(c)}(x) \leq \pi_{b'(c)}(x)$ if $x \in \nu_{b(c)}$, that is,

$$\sum_{i=1}^n \gamma_i \pi_{b_i}(x) \leq \sum_{i=1..n, i \neq k} \gamma_i \pi_{b_i}(x) + \gamma_k \pi_{b'_k}(x).$$

Simplifying this inequality, we have $\gamma_k \pi_{b_k}(x) \leq \gamma_k \pi_{b'_k}(x)$, which implies $x \in \nu_{b_k}$, as required. \square

Next, we generalize this lemma to any possible values of c in Γ . We assume $\gamma_i \neq 0$ for any i , because for any $\gamma_k = 0$,

$$B(c) = \sum_{i=1..n}^{i \neq k} B_i.$$

If $\gamma_i < 0$, the lemma is true if we substitute γ_k with $-\gamma_k$, and $\pi_{b_k}(x)$ with $-\pi_{b_k}(x)$. That means, the lemma remains true if we use $-\pi_{b_k}(x)$ as the distance function, and the furthest-neighbor Voronoi region as ν_{b_k} .

Therefore, the Voronoi cell of $b(c)$ with respect to $B(c)$ is the intersection of the signed Voronoi cells of b_i with respect to B_i , whose signs are determined by the signs of the corresponding γ_i , namely,

$$\nu_{b(c)} = \bigcap_{i=1}^n \nu_{b_i}^{\text{Sign}(\gamma_i)}.$$

This leads to the following Theorem about the intermediate Voronoi complex.

THM. 3.1 The intermediate Voronoi complex is the *superimposition* of all the signed Voronoi complexes, $V_{B_i}^{\text{Sign}(\gamma_i)}$, that is,

$$V(c) = \left\{ \nu_{X(c)} = \bigcap_{i=1}^n \nu_{X_i}^{\text{Sign}(\gamma_i)} \mid \nu_{X(c)} \neq \emptyset, X(c) = \sum_{i=1}^n \gamma_i X_i, X_i \subseteq B_i \right\}.$$

PROOF. \square

According to this theorem, an intermediate Voronoi cell is the collection of the non-empty intersections of n signed Voronoi cells, from the n signed Voronoi complexes of the reference shapes, respectively. See Figure 3.1 for an example of superimposing two signed Voronoi complexes when all the coefficients are positive.

The intermediate Voronoi complexes, $V(c_0)$ and $V(c_1)$, are the same if $\text{Sign}(\gamma_{0,i}) = \text{Sign}(\gamma_{1,i})$ for all i , in which $\gamma_{0,i}$, $\gamma_{1,i}$ are the i -th coordinates of c_0 and c_1 respectively. By this, we can divide Γ into $2^n - 1$ convex partitions with respect to the signs of the coefficients. The number of partitions is $2^n - 1$ rather than 2^n , because the coordinates of a coefficient vector $c \in \Gamma$ can not be all negative. Each partition covers all the values of coefficient vectors, $c = (\gamma_1, \dots, \gamma_n)$, with the same $(\text{Sign}(\gamma_1), \dots, \text{Sign}(\gamma_n))$. For any two coefficient vectors in the same partition, their corresponding intermediate Voronoi complexes are the same.

For a deformation of one intermediate shape into another, its parameterizing path is also divided into finitely many pieces. For all the coefficient vectors lie in a same piece, their corresponding intermediate shapes share a same intermediate Voronoi complex.

4. ALGORITHM FOR THE INTERMEDIATE VORONOI COMPLEXES

An intermediate shape, $\text{body}(B(c))$, is immediate once the intermediate Voronoi complex, $V(c)$, is determined. In this section, we introduce the algorithm to compute $V(c)$.

An intuitive yet time consuming approach is to compute the Voronoi complex of the intermediate weighted point set, $B(c)$. Assume that the cardinality of each weighted point set of the reference shape is no more than m , namely, $\text{card}(B_i) \leq m$, $i = 1..n$. Then, the cardinality of $B(c)$ is no more than m^n . Therefore, the running time of computing the Voronoi complex of $B(c)$ directly is

$$O(nm^n \log m + m^{n \times \lceil d/2 \rceil}) \quad (4)$$

[1, 3, 6].

On the other hand, according to Theorem 3.1, we can compute $V(c)$ by superimposing the signed Voronoi complexes of the reference shapes. We design an algorithm to compute the superimposition of a set of Voronoi complexes. We prove that when $d > 2$, computing $V(c)$ with our algorithm is faster than computing it from $B(c)$ directly.

Since we only care about a set of given Voronoi complexes, in spite of their signs, in the rest of the paper, we simplify $V_{B_i}^{\text{Sign}(\gamma_i)}$ to V_{B_i} , and $\nu_{X_i}^{\text{Sign}(\gamma_i)}$ to ν_{X_i} .

4.1 Superimposing Two Voronoi Complexes

Let V_0 and V_1 be two intermediate Voronoi complexes. In order to generate their superimposition, $V_{0,1}$, we compute all the Voronoi regions in $V_{0,1}$. Each Voronoi region, $\nu_{i,j} \in V_{0,1}$, is the non-empty intersection of a pair of Voronoi regions, $\nu_i \cap \nu_j$, with $\nu_i \in V_0$ and $\nu_j \in V_1$.

We can compute $V_{0,1}$ by a brute force method, namely, testing all the possible pairs of ν_i and ν_j for intersection. Testing whether two Voronoi regions have a non-empty intersection is equivalent to testing whether a set of halfspaces have a common non-empty intersection. We can solve this with linear programming algorithms.

However, we can collect all the Voronoi regions of $V_{0,1}$ with a breadth-first search manner. We denote two Voronoi regions as *neighbors* if they have a common $(d-1)$ -dimensional facet. We starts from a Voronoi region, $\nu_{i',j'} \in V(c)$, which can be computed by locating $\nu_{j'} \in V_1$ which contains a point $p \in \nu_{i'}$. We collect all the neighbors of $\nu_{i',j'}$ in $V_{0,1}$ by collecting all the non-empty intersections of ν_i and the neighbors of ν_j in V_1 , and all the non-empty intersections of ν_j and the neighbors of ν_i in V_0 . For each newly found Voronoi region $\nu_{i,j} \in V_{0,1}$, we collect its neighbors. We repeat this collection iteratively, until no new Voronoi regions are found.

We use an example to illustrate our algorithm. The Voronoi complex $V_{0,1}$ in Figure 1(b) is the superimposition of the Voronoi complexes V_0 in Figure 1(a) and V_1 in 1(c). To construct $V_{0,1}$, we start from $\nu_{14} = \nu_1 \cap \nu_4$. We test whether ν_1 intersects ν_4 's neighbors, ν_5 , ν_6 and ν_7 . Also, we test whether ν_4 intersects ν_1 's neighbors, ν_2 and ν_3 . Then, we get all the four neighbors of ν_{14} in $V_{0,1}$, ν_{17} , ν_{34} , ν_{24} and ν_{15} . Similarly, we can get these neighbors' neighbors, ν_{37} , ν_{36} , ν_{26} and ν_{25} .

The breadth-first search algorithm is output sensitive, the worst case is that we tested all the possible pairs. Since there are at most m Voronoi regions in either V_0 or V_1 , the total number of tests of intersection is at most m^2 . Testing whether two Voronoi regions have non-empty intersection

needs $O(d!m)$ time by the linear programming method [5]. Consequently, the total time complexity of superimposing V_0 and V_1 is $O(d!m^3)$.

4.2 Superimposing n Voronoi Complexes

Next we introduce the algorithm of computing $V(c)$ by superimposing n Voronoi complexes, which is based on the algorithm of superimposing two Voronoi complexes.

For convenience, let $k = \lceil \log_2(n) \rceil$ and $\eta = 2^{k-1}$. We have $\eta < n \leq 2\eta$. We divide the way of superimposing n Voronoi complexes into three cases according to the relationship of n and η . We prove that when $d > 2$, our algorithm can achieve better efficiency than computing the Voronoi complex of $B(c)$ directly.

Case 1:

When $3\eta/2 < n \leq 2\eta$, we superimpose the n Voronoi complexes according to a binary superimposing tree. See Figure 4.1. In the first round, we superimpose V_{2j-1} and V_{2j} for $j = 1.. \eta$. The time complexity for this round is $O(d!2^{k-1}m^3)$. In the second round, we superimpose $V_{(4j-3)(4j-2)}$ and $V_{(4j-1)(4j)}$ for $j = 1.. \eta/2$. In either $V_{(4j-3)(4j-2)}$ or $V_{(4j-1)(4j)}$, there are no more than m^2 Voronoi regions. Thus, the time complexity for the second round is $O(d!2^{k-2}m^{3 \times 2})$. Generally, for the i -th round, the time complexity is $O(d!2^{k-i}m^{3 \times 2^{i-1}})$, $i = 1..k$. After k rounds of such superimposition, we can get $V(c)$. Therefore, the total time complexity to superimpose the n Voronoi complexes, V_1, V_2, \dots and V_n , is

$$O\left(\sum_{i=1}^k (d!2^{k-i}m^{3 \times 2^{i-1}})\right).$$

Since m is greater than 2 in practice, we have $(d!2^{k-i_1}m^{3 \times 2^{i_1-1}}) < (d!2^{k-i_2}m^{3 \times 2^{i_2-1}})$ for any $i_1 < i_2$. Thus, the total time complexity is equivalent to $O(kd!m^{3 \times 2^{k-1}})$. For a fixed d , we can ignore the $d!$ and only consider the time complexity as $O(km^{3 \times 2^{k-1}})$. Adding in the time of computing V_1, \dots, V_n , the overall time complexity to compute $V(c)$ is

$$O(km^{3 \times 2^{k-1}} + nm \log m + nm^{\lceil d/2 \rceil}). \quad (5)$$

When $d > 2$ and $n > 1$, the time complexity of the intuitive way in Equation 4 is dominated by $O(m^{n \times \lceil d/2 \rceil})$, and the time complexity of our algorithm in Equation 5, is dominated by $O(km^{3 \times 2^{k-1}} + nm^{\lceil d/2 \rceil})$. In practice, we can assume that n, k , and d are smaller than m . Under such assumption, our algorithm has better efficiency if we can show that $m^{3 \times 2^{k-1}} < m^{n \lceil d/2 \rceil}$. Since $n > 3\eta/2$ and $d > 2$, we have

$$m^{3 \times 2^{k-1}} = m^{3 \times \eta} < m^{2n} \leq m^{n \lceil d/2 \rceil},$$

as required.

Case 2:

When $\eta/8 < n \leq 3\eta/2$, we make some adjustment on the organization of the superimposing tree. The adjusted superimposing tree is as Figure 4.2. In the first round, we superimpose V_{2j-1} and V_{2j} for $j = 1.. \eta/2$. The time complexity for this round is $O(d!2^{k-2}m^3)$. In the second round,

we superimpose $V_{(2j-1)(2j)}$ and $V_{\eta+j}$ for $j = 1.. \eta/2$. The time complexity for this round is $O(d!2^{k-2}m^{3 \times 2})$. From the third round to the k -th round, we superimpose the resulting $\eta/2$ Voronoi complexes with the binary superimposing tree introduced in the first case. Thus, the total time complexity is

$$O(d!2^{k-2}m^3 + d!2^{k-2}m^{3 \times 2} + \sum_{i=3}^k (d!2^{k-i}m^{3 \times 3 \times 2^{i-3}})),$$

which is equivalent to $O(kd!m^{3 \times 3 \times 2^{k-3}})$. Similarly to the first case, our algorithm has better efficiency if we can show that $m^{3 \times 3 \times 2^{k-3}} < m^{n \lceil d/2 \rceil}$. Since, $n > 9\eta/8$ and $d > 2$, we have

$$m^{3 \times 3 \times 2^{k-3}} = m^{9\eta/4} < m^{2n} \leq m^{n \lceil d/2 \rceil},$$

as required.

Case 3:

The final case is when $\eta < n \leq 9\eta/8$. Since both n and η are integers, we know that $\eta \geq 8$. We adjust the superimposing tree as in Figure 4.3. We first superimpose $V_{8j-7}, V_{8j-6}, \dots, V_{8j}$ for $j = 1.. \eta/8$ with the binary superimposing tree. The time complexity is $O(\sum_{i=1}^3 (d!2^{k-i-1}m^{3 \times 2^{i-1}}))$. Then, we get $\eta/8$ Voronoi complexes, $V_{1..8}, V_{9..16}, \dots, V_{(\eta-7).. \eta}$. In the next round, we superimpose $V_{(8j-7)..(8j)}$ and $V_{\eta+j}$ for $j = 1.. \eta/8$. The time complexity is $O(d!2^{k-4}m^{3 \times 8})$. At last, we superimpose the resulting $\eta/8$ Voronoi complexes with the binary superimposing tree. Consequently, the total time complexity is

$$O\left(\sum_{i=1}^3 (d!2^{k-i-1}m^{3 \times 2^{i-1}}) + d!2^{k-4}m^{3 \times 8} + \sum_{i=5}^k (d!2^{k-i}m^{3 \times 9 \times 2^{i-5}})\right),$$

which is equivalent to $O(kd!m^{3 \times 9 \times 2^{k-5}})$. Similarly to the former two cases, our algorithm has better efficiency if we can show that $m^{3 \times 9 \times 2^{k-5}} < m^{n \lceil d/2 \rceil}$. Since, $n > \eta$ and $d > 2$, we have

$$m^{3 \times 9 \times 2^{k-5}} = m^{27\eta/16} < m^{2\eta} < m^{2n} \leq m^{n \lceil d/2 \rceil},$$

as required.

In this section, we introduced a breadth-first algorithm to superimpose two Voronoi complexes. Then, for n Voronoi complexes, we discussed three different cases according to the value of n . For each individual case, we proposed the way of superimposing n Voronoi complexes, and proved that our algorithm has better efficiency than the intuitive way. Since these three cases cover all the possibilities, our algorithm is faster than the intuitive way in computing $V(c)$ when $d > 2$.

5. CONCLUSION

In this paper, we designed a new algorithm to compute the Voronoi complex of an intermediate shape basing on Theorem 3.1. We prove that when $d > 2$, our algorithm is faster than computing the Voronoi complex from $B(c)$ directly. Moreover, since all the intermediate shapes share finitely many common Voronoi complexes, we are able to compute intermediate shapes in real time, by reusing the generated common Voronoi complexes. This makes it possible to generate, visualize, and customize shape deformations.

About the future direction, we want to save the time complexity of superimposing n Voronoi complexes by rearranging the order of the pairwise superimpositions. We may solve this problem with dynamic programming. The goal is to change the superimposing order so that the cardinalities of the intermediate superimposing results are as small as possible.

6. REFERENCES

- [1] CHAZELLE, B. An optimal convex hull algorithm in any fixed dimension. In *Discrete Comput. Geom.* (1993), pp. 10:377–409.
- [2] CHENG, H., EDELSBRUNNER, H., AND FU, P. Shape Space from Deformation. *Proc. 6th Pacific Conf. Comput. Graphics Appl.* (1998), 104–113.
- [3] CLARKSON, K. L., AND SHOR, P. W. Applications of random sampling in computational geometry, II. *Discrete and Computational Geometry 4*, 1 (1989), 387–421.
- [4] EDELSBRUNNER, H. Deformable Smooth Surface Design. *Discrete Comput. Geom* 21 (1999), 87–115.
- [5] SEIDEL, R. Linear programming and convex hulls made easy. In *Proc. 6th Annu ACM Sympos. Coput. Geom.* (1990), pp. 211–215.
- [6] SEIDEL, R. Small-dimensional linear programming and convex hulls made easy. In *Discrete Comput. Geom.* (1991), pp. 6:423–434.

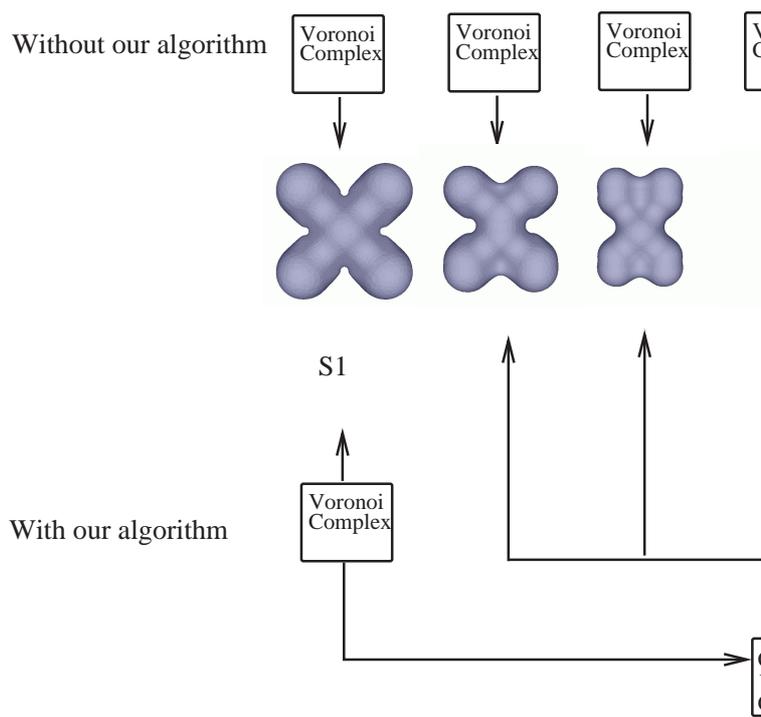


Figure 1.1: A deformation of the shape $S1$ into the shape $S2$ in \mathbb{R}^3 . In the intuitive way, we need to compute the Voronoi complex for each intermediate shape individually. However, with our algorithm, we only need to generate one common Voronoi complex and reuse it.

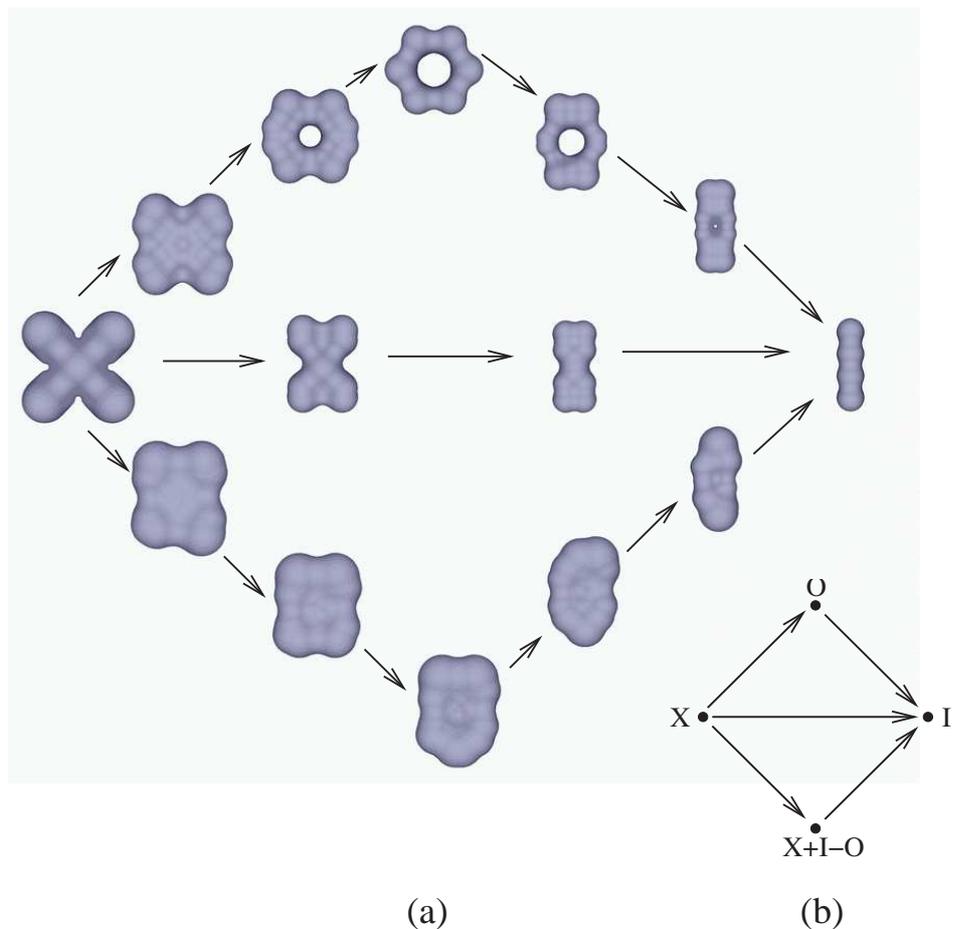


Figure 1.2: Three deformations of 'X' into 'I'. The middle path represents the direct deformation. The top sequence is mixing 'O' into the deformation. The bottom sequence is avoiding 'O' in the deformation. The concept is demonstrated in Figure (b). All these shapes are skin bodies in \mathbb{R}^3 .

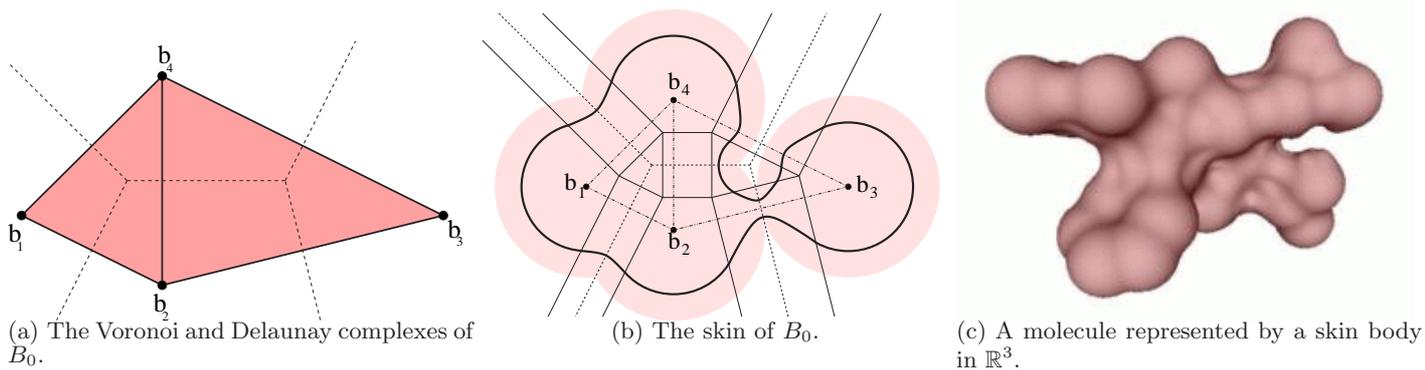


Figure 2.1: The left subfigure is the Voronoi and Delaunay complexes of a weighted point set in \mathbb{R}^2 , $B_0 = \{b_1, b_2, b_3, b_4\}$. The center subfigure is the skin of B_0 , which is constructed on the complexes of B_0 . The right subfigure is the skin body of a molecule in \mathbb{R}^3 space.

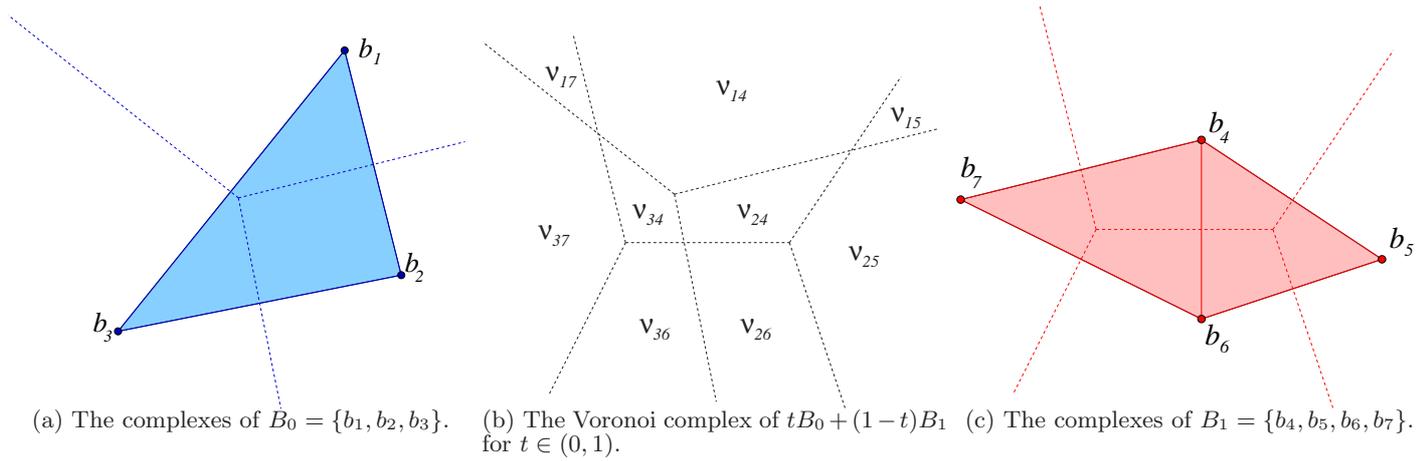


Figure 3.1: Superimposition of two signed Voronoi complexes.

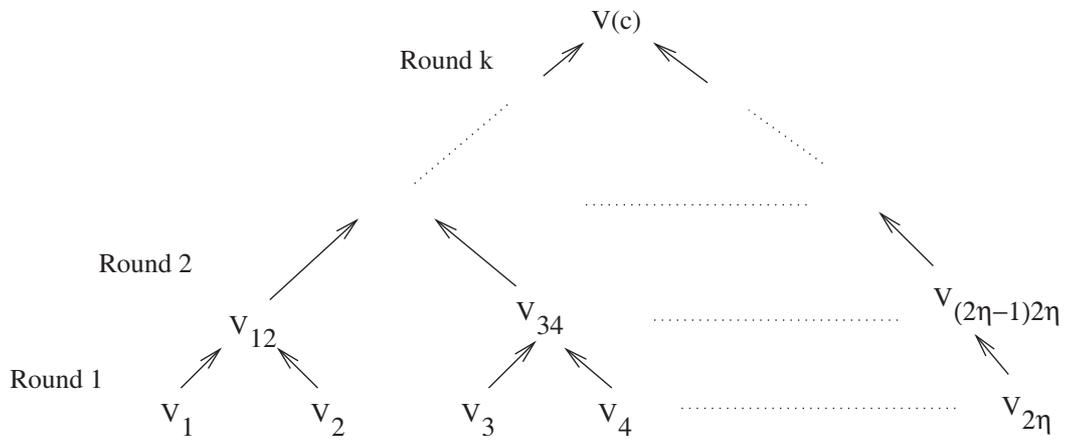


Figure 4.1: Superimposing n Voronoi complexes when $3\eta/2 < n \leq 2\eta$.

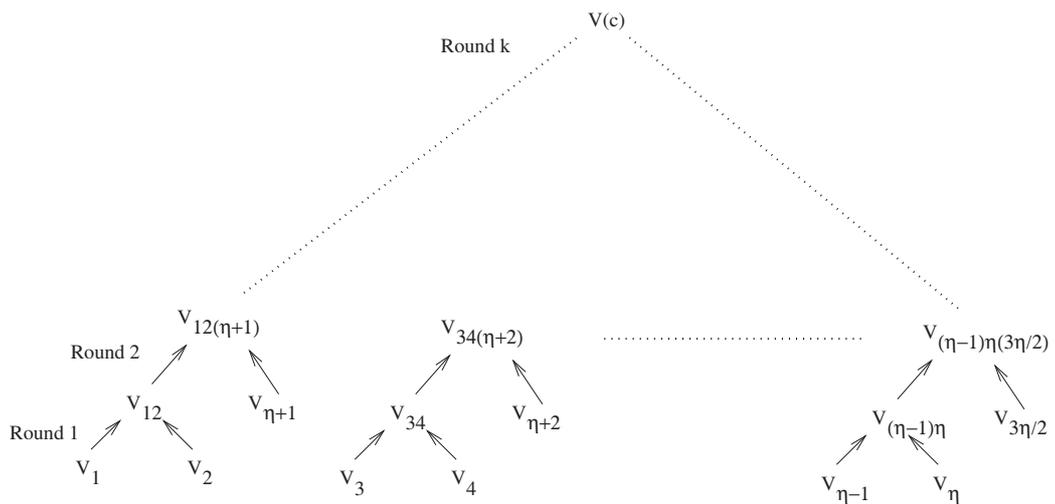


Figure 4.2: Superimposing n Voronoi complexes when $9\eta/8 < n \leq 3\eta/2$.

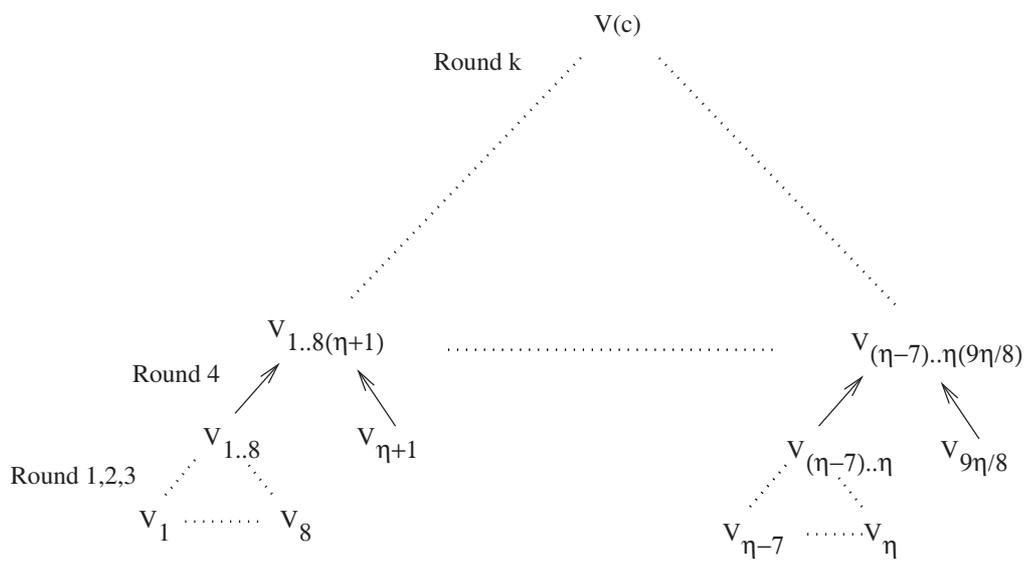


Figure 4.3: Superimposing n Voronoi complexes when $\eta < n \leq 9\eta/8$.