SAMPLE SHUFFLING FOR QUALITY HIERARCHIC SURFACE MESHING

Tamal K. Dey Joachim Giesen James Hudson [1]

Department of CIS, Ohio State University, Columbus, OH 43210.

ABSTRACT

A variety of applications need to compute a surface mesh from a set of discrete sample points obtained by some scanning device, or generated by some scientific simulations. A mesh hierarchy either for multigrid methods or mutiresolution modeling can be created from such samples by first generating a sample hierarchy and then computing surface meshes from them. This approach preserves geometric and topological features automatically and avoids self-intersection, a major concern in the edge collapsing methods used for mesh decimation. We introduce the sample shuffling technique to create the sample hierarchy. It deletes and repositions the sample points so that the triangles have bounded aspect ratio in a resulting mesh out of them.

Keywords: Sampling, decimation, surface mesh, quality meshing, multiresolution, reconstruction.

1. INTRODUCTION

A wide range of application domains and the modern advancement in laser technology have sparked new developments in modeling surfaces from samples. In a typical application the surface of a physical object is scanned with a scanning device such as laser range scanner that generate a dense set of discrete samples. A surface mesh approximating the original surface is created with the sample points as vertices. This problem, popularly known as surface reconstruction, has been a topic of active research in recent years and a number of algorithms with various guarantees and capabilities have been designed [1, 3, 4, 5, 12].

Surface meshes are used for various purposes such as displays in computer graphics, finite element methods in physical simulations or in computer aided manufacturing. Often large surface meshes become unwieldy for these postprocessings. Naturally, a demand arose to simplify these surfaces in a meaningful way. A popular approach, originally proposed by Hoppe et. al [16], contracts edges of the triangulation to produce simplified meshes. Extra care is needed to ensure that edge contraction does not cause any major change in shape. This means topological and geometric features of the shape should be preserved during edge contractions. Both of these conditions can be achieved by combining the topology preserving edge contractions of [11] and the quadric error metric of [15]. However, these edge contraction methods tend to produce skinny triangles with poor aspect ratio. Poor quality meshes pose a serious problem in many applications which has triggered a considerable research on quality mesh generation [6, 9, 14, 17, 18, 19, 20, 22, 23, 24]. Moreover, edge contractions can introduce self-intersection warranting a geometric validity check, which adds more complications.

In this paper we develop an alternative approach of creating the mesh hierarchy by first producing a hierarchy of samples and then computing surface meshes from them. The advantage of this approach is that one does not need to worry about

[1] This research is supported by NSF grant CCR-9988216.
e-mail: {tamaldey,giesen,jhudson}@cis.ohio-state.edu

self-intersection since the output mesh is a subcomplex of a Delaunay triangulation. If we make sure that the samples in the hierarchy are sufficiently dense for a meaningful reconstruction, topological and geometric features are automatically preserved. In addition, we introduce the technique of *sample shuffling* that guarantees a bounded aspect ratio for an output mesh. This is reminiscent of smoothing techniques that are usually applied on volume meshes [2, 18, 19]. If the surface is known, the samples can be chosen carefully to satisfy quality mesh criteria as done in [7, 8]. The fact that the surface is unknown makes sample shuffling harder.

Sample shuffling incorporates both deletion and insertion of points. A sample point p is deleted if it is too close to its nearest neighbor, and sometimes is replaced with another point if deletion of p destroys the sampling density below a threshold. Deletions ensure that there is a lower bound on the interpoint distances. Insertions, on the other hand, ensure that the triangles in the output mesh have upper bound on their circumradii. These two conditions guarantee a bounded aspect ratio for the triangles in the output mesh.

2. OVERVIEW AND DEFINITIONS

Let P be a set of sample points from a smooth surface $S \subset \mathbb{R}^3$ without boundary. Our goal is to create a hierarchy of triangular meshes each approximating S. Each mesh is created by shuffling P to a new point set Q and then applying a Delaunay based surface meshing technique called COCONE which improved the CRUST algorithm pioneered by Amenta and Bern [1].

2.1 Sampling

Obviously, the sample P needs to be sufficiently dense to carry enough information about the features of S. Mathematical definition of dense sampling is introduced by Amenta and Bern in [1] which we call ε-sampling.

The *medial axis* of a smooth surface $S \subset \mathbb{R}^3$ is the closure of the set of points that have two or more nearest points on S. The *local feature size* is given by the function $f : S \to \mathbb{R}$ where $f(p)$ is the least distance of $p \in S$ to the medial axis. The function $f()$ is 1-Lipschitz continuous, i.e., $f(p) \leq f(q) + ||p - q||$ for any two points p, q in S. A point set $P \subset S$ is called an ε-sample of S if each point $p \in S$ has a sample within distance of $\varepsilon f(p)$.

2.2 Mesh hierarchy

The mesh hierarchy is created from a sequence of point sets $\{Q\}$ that are computed from P. Each point set Q is generated by decimating and moving points in P using two parameters ρ and ρ' whose roles are explained later.

Voronoi diagram and its dual Delaunay triangulation are the basic data structures in our algorithms. We denote the Voronoi diagram and the Delaunay triangulation of a point set $P \subset \mathbb{R}^3$ by V_P and D_P respectively. The Voronoi cell of a point $p \in P$ is denoted as V_p. The notation $\angle(\mathbf{u}, \mathbf{v})$ denotes the *acute* angle between the lines supporting two vectors $\mathbf{u}$ and $\mathbf{v}$.

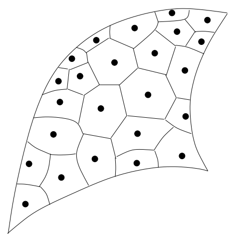

Figure 1. Intersection of V_P with the surface S

The sample hierarchy $\{Q\}$ is generated by detecting and deleting the sample points that lie in oversampled regions. Consider the intersection of the Voronoi diagram V_P with S which creates the restricted Voronoi diagram on S. Each of the restricted Voronoi cell $V_{p,S} = V_p \cap S$ is a topological disk if P is a dense sample. See Figure 1 for an illustration. Let d_p be the distance of the farthest point in $V_{p,S}$ from p. All points in $V_{p,S}$ have p as nearest sample. Thus, the point p lies in an oversampled region, if d_p is small compared to the local feature size $f(p)$. A parameter ρ can be used to differentiate between "small" and "large". If p is detected to lie close to its nearest neighbor, it is deleted. If in addition, p lies in an oversampled region, no point is inserted to replace p. Otherwise, a new point replaces p. Clearly, the value of ρ determines the level of decimation creating the hierarchy.

Unfortunately, we cannot perform these comparisons exactly since S is unknown. Instead, we approximate $V_{p,S}$ by *cocone* and d_p by its *radius* which are defined below. The local feature size $f(p)$ is estimated with the "poles" as introduced in [1].

Poles: The farthest Voronoi vertex p^+ in V_p is called the positive *pole* of p. The negative pole of p is the farthes point $p^- \in V_p$ making an angle more than $\frac{\pi}{2}$ with the vector $\mathbf{v}_p = p^+ - p$, the *pole vector* for p. If V_p is unbounded, p^+ is taken at infinity, and the direction of $\mathbf{v}_p$ is taken as the average of all directions given by unbounded Voronoi edges.

Cocone: The set $C_p(\theta, \mathbf{v}) = \{y \in V_p : \angle((y-p), \mathbf{v}) \geq \frac{\pi}{2} - \theta\}$ is called the cocone of p with axis $\mathbf{v}$ and angle θ. In words, $C_p(\theta, \mathbf{v})$ is the complement of a double cone (clipped within V_p) centered at p with an opening angle $\frac{\pi}{2} - \theta$ around the axis aligned with $\mathbf{v}$. Usually, we take $\mathbf{v} = \mathbf{v}_p$, the *pole* vector at p and $\theta \leq \frac{\pi}{8}$. See Figure 2 for an illustration of cocones in two and three dimensions.

Radius and height: The *radius* for a point p, r_p, is defined as $r_p = \max\{||p-y|| : \angle((y-p), \mathbf{v}) = \frac{\pi}{2} - \theta\}$. Define the *height* h_p as $||p^- - p||$. Obviously, $h_p \leq ||p^+ - p||$.

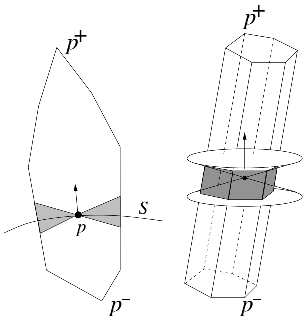

Figure 2. A 2D Voronoi cell (left) and a 3D Voronoi cell (right) together with the normalized pole vector and the cocone (shaded).

2.3 Aspect Ratio

The ratio of the circumradius of a triangle over its shortest edge length measures its aspect ratio. In order to bound this ratio from above we need to ensure a lower bound on the shortest edge length and an upper bound on the circumradius.

The triangles generated by the surface meshing algorithm COCONE have circumradii less than the radius r_p where p is any of their vertices. The radius r_p has a bound $O(\rho)f(p)$ if the sample is a ρ-sample. Thus, ensuring that the point set is not too sparse guarantees an upper bound on the circumradii of the triangles. This condition is required in any case for a fruitful surface meshing. However, ρ-sampling does not prevent the sample points to be arbitrarily close and thus cannot prohibit arbitrarily small edges. So, we check if a sample point p is close to any other sample point below a threshold. If so, we delete p. However, deleting p may ruin the sampling density. If that is the case, we insert a new point $\hat{p}$ sufficiently away from all other existing sample points. This has the effect of shuffling p to a new position $\hat{p}$. This shuffling is the key to maintaining both the sampling density up to a level guided by an input parameter ρ and also a lower bound on inter point distances guided by another input parameter $\rho' < \rho$.

2.4 Main result

Our main result is that a dense sample of a surface can be shuffled to a new point set so that a surface mesh can be computed from it which is homeomorphic and geometrically close to the original surface. Moreover, each triangle has a bounded aspect ratio.

The proofs of the guarantees are based on an asymptotic analysis which assumes that input sample is extremely dense and the target density in the hierarchy are extremely high, i.e., ε, ρ and ρ' are very small. It is important to note that the constants hidden in the asymptotic notations depend on the sampled surface. However, they are independent of ε, ρ and ρ'. Thus, for a given surface our asymptotic analysis holds for a sufficiently small ε, ρ, ρ'.

We adopted the asymptotic analysis to ease the onus of calculating the constants rigorously. These constants are irrelevant for most practical data anyway. Nevertheless, the proofs with those hidden constants embody the essential ingredients needed for the correctness of the algorithm and, in fact, guide its design. The practical validity of the algorithm is exhibited by our empirical results, where the values of these constants need not be so small.

Theorem 1. For sufficiently small ε, ρ and ρ' with $\varepsilon < \rho' < \rho < 1$, an ε-sample P of a smooth surface S can be shuffled to a new point set Q such that a surface mesh M can be computed from Q with the following properties:
a. M is homeomorphic to S,
b. any point on M is $O(\rho)f(p)$ away from some $p \in S$,
c. each triangle has a bounded aspect ratio $O(\frac{\rho}{\rho'})$.

3. SAMPLE SHUFFLING

3.1 Algorithm

The shuffling algorithm deletes a sample point with a close nearest neighbor and sometimes replaces them with new points. All decisions are made in a scale independent manner relative to h_p that estimates $f(p)$ as stated in Lemma 3.1.

Deletions: Let δ_p be the distance of p to its nearest neighbor q. We use the ratio $\frac{\delta_p}{h_p}$ to determine if p is close to any existing point. If any sample p has $\frac{\delta_p}{h_p} < \rho'$, we delete p, where $\rho' < \rho$ is an user chosen parameter.

Insertions: If we continuously delete points since they have close neighbors, the point set may become locally sparse. To prevent this we check if $\frac{r_p}{h_p} > \frac{\rho}{2}$. If this condition holds, we know that $V_{p,S}$ has a point far away from p in a relative sense and we insert a point close to this far away point. This repairs the damage on sample density caused by the deletion of p, and at the same time ensures that the new point is not added close to any other existing point. We use $\hat{p} \in C_p(\theta, \mathbf{v}_p)$, where $||p - \hat{p}|| = r_p$ as the new point to insert. It turns out that insertion of $\hat{p}$ is not enough since deletion of p may increase the radii for other samples. This triggers some more insertions.

SHUFFLE(P, ρ, ρ')
1 **while** $\exists p \in P \cap S$ do
2   **if** $\frac{\delta_p}{h_p} < \rho'$ **then**
3   delete p from P;
4   **if** $\frac{r_p}{h_p} > \frac{\rho}{2}$ insert $\hat{p}$ into P;
5   update V_P;
6   **while** $\exists p \in P \cap S$ with $\frac{r_p}{h_p} > \rho$ do
7   insert $\hat{p}$ into P and update V_P;
8   **endwhile**
9   **endif**
10 **endwhile**
11 **return** P

It is important to notice that the condition in the *while* loops in steps 1 and 6 ensure that only points from original sample may trigger a deletion or insertion. It turns out that these restricted deletions and insertions are sufficient for our purpose.

The height h_p is computed from the Voronoi diagram V_P. As P is shuffled, V_P changes. We do not recompute h_p. The reason is that poles approximate the medial axis better with denser sample than the sparser ones; and we use h_p to approximate the distance of p to the medial axis. This also means that the cocone axis remains same for a point p throughout the shuffling procedure. However, the radius r_p changes as V_P changes.

3.2 Geometric guarantees

Let Q be the point set output by SHUFFLE. We establish two important geometric facts about the output Q.

Fact 1. All new points are inserted at a safe distance from all other points. Also, original sample points with close neighbors are deleted. This ensures a lower bound on the interpoint distances.

Fact 2. The algorithm SHUFFLE always maintains a point set, say Z, that has a nearby sample Z^* on S and Z^* is a dense sample of S. Formally, Z^* is defined as $Z^* = \{z^*\}$ where $z^* \in S$ is the nearest point to $z \in Z$ on S.

Other than quality guarantees, the above two facts imply that the Voronoi cell of each $p \in Q$ is long and thin along the normal direction of a nearby point in S. This is an essential ingredient for our surface meshing algorithm. The proofs of Fact 1 and Fact 2 need each other. We break this circular dependence by considering only a subset of inserted points in Lemma 3.6 that establishes Fact 2. Fact 1 is proved with a sequence of lemmas below.

First, we need some results from earlier works. Recent results in [4, 5] show that poles approximate the medial axis. Thus, the height h_p approximates a distance to a point on the medial axis. We can take minimum among all distances to the poles as an approximation to $f(p)$. Instead, we take h_p as an approximation to $f(p)$. If one examines carefully, this approximation is justified in the asymptotic framework as stated in the next lemma. We emphasize that the big-O notation hides some surface dependent constants independent of ε, ρ and ρ'.

Lemma 3.1. *Let P be an ε-sample of a smooth surface S. Then, for any $p \in P$, $h_p \leq \lambda f(p)$ where $\lambda = 1 + O(\varepsilon)$ can be chosen as a constant.*

Lemma 3.2 and Lemma 3.3 are taken from [1] and [3] respectively. We state them again in asymptotic notations hiding the surface constants.

Lemma 3.2. *Let P be an ε-sample of a smooth surface S. Let $\boldsymbol{v}$ be a vector from a sample p to one of its poles. Then, $\angle(\boldsymbol{v}, \boldsymbol{n}_p) = O(\varepsilon)$.*

Lemma 3.3. *Let $\boldsymbol{v}$ be a vector with origin at p so that $\angle(\boldsymbol{v}, \boldsymbol{n}_p) = O(\varepsilon)$. Then there exists a constant c so that if P is an $O(\rho)$-sample and $\theta = c\rho$, a small angle, then (a) any point $x \in V_{p,S}$ satisfies $\angle((x-p), \boldsymbol{n}_p) \geq \frac{\pi}{2} - O(\rho)$, (b) $V_{p,S}$ lies inside $C_p(\theta, \boldsymbol{v})$, and (c) $r_p = O(\rho)f(p)$.*

Now, we show that Fact 1 holds for all points $q = \hat{p}$ inserted in step 4.

Lemma 3.4. *For any point $q = \hat{p}$ inserted in step 4, we have $||p - q|| = \Theta(\rho)f(q^*)$ and $||q - q^*|| = O(\rho^2)f(q^*)$.*

Proof. The upper bound for the distance $||q-p||$ is enforced by the inner `while` loop (steps 6-8). It ensures that no original sample point p can have $\hat{p}$ more than $\rho r_p = O(\rho)f(p)$ away from p for the next iteration of the outer `while` loop. The lower bound follows from the `if` condition in step 4.

To prove $||q - q^*|| = O(\rho^2)f(q^*)$ consider the line segment qy where y is the mirror image of q with respect to the tangent plane π_p at p. Since the surface is locally flat, all points in S in the triangle pqy make a small angle $\alpha = O(\rho)$ with π_p. Also, with a proper choice of the cocone angle $\theta = O(\rho)$ one can guarantee that pq makes an angle more than α with π_p which implies that S intersects qy. We have $||q-y|| = 2||p-q|| \sin \frac{\angle qpy}{2} = O(\rho)f(p) \cdot O(\rho)$ and $||q - q^*|| \leq ||q-y|| = O(\rho^2)f(p)$. Using the Lipschitz property of $f(\cdot)$ we can show that $f(q^*) = \Theta(f(p))$ completing the proof. □

Now let us consider the points that are inserted in step 7. Any point $q = \hat{p}$ inserted in this step has a lower bound on the distance $||q-p||$ due to the condition in step 6 which leads to the following lemma.

Lemma 3.5. *For any point $p \in Q \cap S$ we have $||p-q|| > \frac{\rho\lambda}{2}f(p)$ where q is a point inserted in step 7.*

We need more complicated argument to establish the upper bound. The fact that all new points are inserted near S but sufficiently away from all other points establishes that the Voronoi cells remain long and thin and the cocone radii have upper bounds. Lemma 3.6, 3.7 and 3.8 establish these facts which lead to Lemma 3.9. We provide their proofs in the appendix. Unfortunately, the reasoning that a new inserted point $q = \hat{p}$ lies near S needs an upper bound on its distance from p. This brings us to a circular reasoning. We break this circularity by considering only the points inserted in step 4 in the following lemmas.

Lemma 3.6. *Let Z denote the set of samples not deleted so far together with the points inserted so far in step 4 at any time of the algorithm* Shuffle. *Then Z^* is $O(\rho)$-sample of S.*

Lemma 3.7. *Let $Z \subset \mathbb{R}^3$ be a set of points such that for any two points p, q where $p \in Z \cap S$ and $q \in Z \setminus S$, we have $||q - q^*|| = O(\rho^2)f(q^*)$ and $||p - q|| = \Omega(\rho f(p))$. Then, there exist two empty tangent balls at p with radius $\Omega(f(p))$.*

Lemma 3.8. *Let Z be a point set that satisfies the conditions stated in the previous lemma and Z^* be an $O(\rho)$-sample of S. Let $p \in Z \cap S$ and $\boldsymbol{v}$ be a vector with $\angle(\boldsymbol{v}, \boldsymbol{n}_p) = O(\epsilon)$. Then, the radius r_p of the cocone $C_p(\theta, \boldsymbol{v})$ is $O(\rho)f(p)$, where $\theta = O(\rho)$.*

Lemma 3.9. *For any point $q = \hat{p}$ inserted in step 7, we have $||p - q|| = \Theta(\rho)f(q^*)$ and $||q - q^*|| = O(\rho^2)f(q^*)$.*

Now we are ready to prove Fact 1.

Lemma 3.10. *For any two points p, q in Q we have $||p - q|| = \Omega(\rho')\min\{f(p^*), f(q^*)\}$.*

Proof. First assume that $p = p^* \in P$. In that case, the condition in the outer *while* loop ensures that p is at least $\rho' h_p = \Omega(\rho')f(p)$ away from any other point $q \in Q$ satisfying the lemma. If $p \notin P$, either step 4 or 7 has inserted it. Without loss of generality, assume that q is already present in the point set when p is inserted. In that case $p = \hat{s}$ is at least $\rho h_s = \Omega(\rho)f(s)$ from q. But, Lemma 3.4 and Lemma 3.9 imply that $||p^* - s|| = O(\rho)f(s)$. Applying Lipschitz condition on $f(\cdot)$, we get that $f(s) = \Omega(f(p^*))$. Thus, $||p - q|| = \Omega(\rho)f(s) = \Omega(\rho)f(p^*) = \Omega(\rho')f(p^*) = \Omega(\rho')\min\{f(p^*), f(q^*)\}$. □

4. SURFACE MESHING

4.1 Algorithm

We employ the Cocone algorithm of [3, 10] to compute a triangulated surface M from Q, the output of Shuffle.

Cocone (Q, θ)
compute the Voronoi diagram V_Q of Q.
$T := \emptyset$
for all $q \in Q$
  $T := T \cup$ CandidateTriangles(q)
endfor
extract a manifold M from T.
return M

The algorithm CANDIDATETRIANGLES(q) returns the set of triangles dual to Voronoi edges intersecting the cocone $C_q(\theta, \mathbf{v}_q)$. The manifold extraction step first removes triangles that have a bare edge, i.e. no other triangle is incident to that edge. Such removals are necessary for a subsequent walk over the remaining candidate triangles as described in [1].

The output triangulation M has the topological guarantee that the underlying space of M is homeomorphic to S; see [3] for details. Also, it has the geometric guarantee that the Hausdorff distance between M and S is small.

4.2 Topological guarantee

The topological guarantee in COCONE was proved in [3] using the following two properties of the set of candidate triangles T.

Property 1. Any triangle $t \in T$ has a small circumradius of $O(\rho)f(p^*)$ where p is any vertex of t.

Property 2. Candidate triangles include the Delaunay triangles that are dual to the Voronoi edges intersecting the surface S.

Since Q is not necessarily a sample of S, we cannot use the proofs of [3] directly. We need separate arguments to establish the above properties. Lemma 4.3 and 4.4 ensure Property 1 and 2 respectively. The proofs of the next two lemmas and Lemma 4.4 are given in the appendix.

Lemma 4.1. *Let $p \in Q$ be any point output by* SHUFFLE. *There exists two balls of radii $\Omega(f(p^*))$ touching at p tangentially.*

Lemma 4.2. *Let x be a point in $V_p \in V_Q$ so that $||p-x|| = \Omega(f(p^*))$. Then $\angle(\boldsymbol{n}_{p^*}, (x-p)) = O(\rho)$.*

Lemma 4.3. *Let $\boldsymbol{v}$ be a pole vector at $p \in Q$. The cocone $C_p(\frac{\pi}{8}, \boldsymbol{v})$ has a radius $O(\rho)f(p^*)$.*

Proof. Again consider the two balls B and B' touching at p with radii $\Omega(f(p^*))$ as stated in Lemma 4.1. Let x be the point which is farthest from p in $C_p(\frac{\pi}{8}, \mathbf{v})$ and x, m' lie on opposite sides of S. It follows from Lemma 4.2 that $\angle(n_{p^*}, \mathbf{v}) = O(\rho)$. Since $\angle(\mathbf{v}, (x-p)) \geq \frac{3\pi}{8}$ we have $\angle xpm \geq \frac{3\pi}{8} - O(\rho)$. Using similar argument as in Lemma 4.2 one can show that $\angle pm'x = O(\rho)$. Thus, $\angle pxm' = \angle xpm - \angle pm'x > \frac{3\pi}{8} - O(\rho)$. Let py be the perpendicular from p to the line of xm'. It follows that $||p-x|| = ||p-y||/\sin(\angle pxm') = O(\rho)f(p^*)$ proving the claim. $\square$

Lemma 4.4. *Let e be a Voronoi edge in $V_p \in V_Q$ intersected by S. The point $e \cap S$ is in $C_p(\frac{\pi}{8}, \boldsymbol{v})$.*

4.3 Mesh distance

The output mesh M should be geometrically close to the sampled surface S. We show that each point on the surface S has a nearby point on the output mesh. Since each triangle is small (Property 1), it follows that each point on the output mesh also has a nearby point on the surface. Recall that S is partitioned into restricted Voronoi cells by the Voronoi diagram V_Q. Next lemma shows that each restricted Voronoi cell has a nearby vertex in M.

Lemma 4.5. *For any point $p \in Q$, $V_{p,S}$ is contained in a ball of radius $O(\rho)f(p^*)$.*

Proof. It follows from Lemma 3.6 that Q^* is an $O(\rho)$-sample of S. If any $x \in V_{p,S}$ is nearer to p^* than any other point in Q^*, we have $||x-p^*|| = O(\rho)f(p^*)$. Otherwise, let $q^* \neq p^*$ be the nearest to x. Then, $||q-x|| \leq ||q-q^*|| + ||q^*-x||$. Also, x is nearer to p than any point in Q and hence $||p-x|| \leq ||q-x||$. Combining these inequalities we get $||p^*-x|| \leq ||p-p^*|| + ||p-x|| \leq ||q^*-x|| + ||q-q^*|| + ||p-p^*||$. But, since Q^* is an $O(\rho)$-sample of S, we have $||q^*-x|| = O(\rho)f(x)$. Thus, $||p^*-x|| = O(\rho)f(x) + O(\rho^2)f(q^*) + O(\rho^2)f(p^*) = O(\rho)f(p^*)$. Since $||p-p^*|| = O(\rho^2)f(p^*)$, the lemma follows. $\square$

4.4 Mesh guarantees

Now we have all the ingredients for the guarantees in Theorem 1. Since any triangle t in T is dual to Voronoi edges intersected with a cocone, its circumcircle has a radius $O(\rho)f(p^*)$ where p is a vertex of t (Lemma 4.3). This ensures Property 1 for candidate triangles. Lemma 4.4 assures Property 2. Argument in [3] can be extended to show that the output surface M is homeomorphic to S.

The shortest edge in t has a length $\Omega(\rho')f(p^*)$ (Lemma 3.10). Circumradius of any triangle $t \in T$ has a radius $O(\rho)f(p^*)$ where p is any of its vertex. Hence the aspect ratio of t is at most $O(\frac{\rho}{\rho'})$.

5. EXPERIMENTS

We implemented SHUFFLE on a 733 Mhz PC with 512 MB memory. We used CGAL [26] to compute the Delaunay triangulation with filtered floating

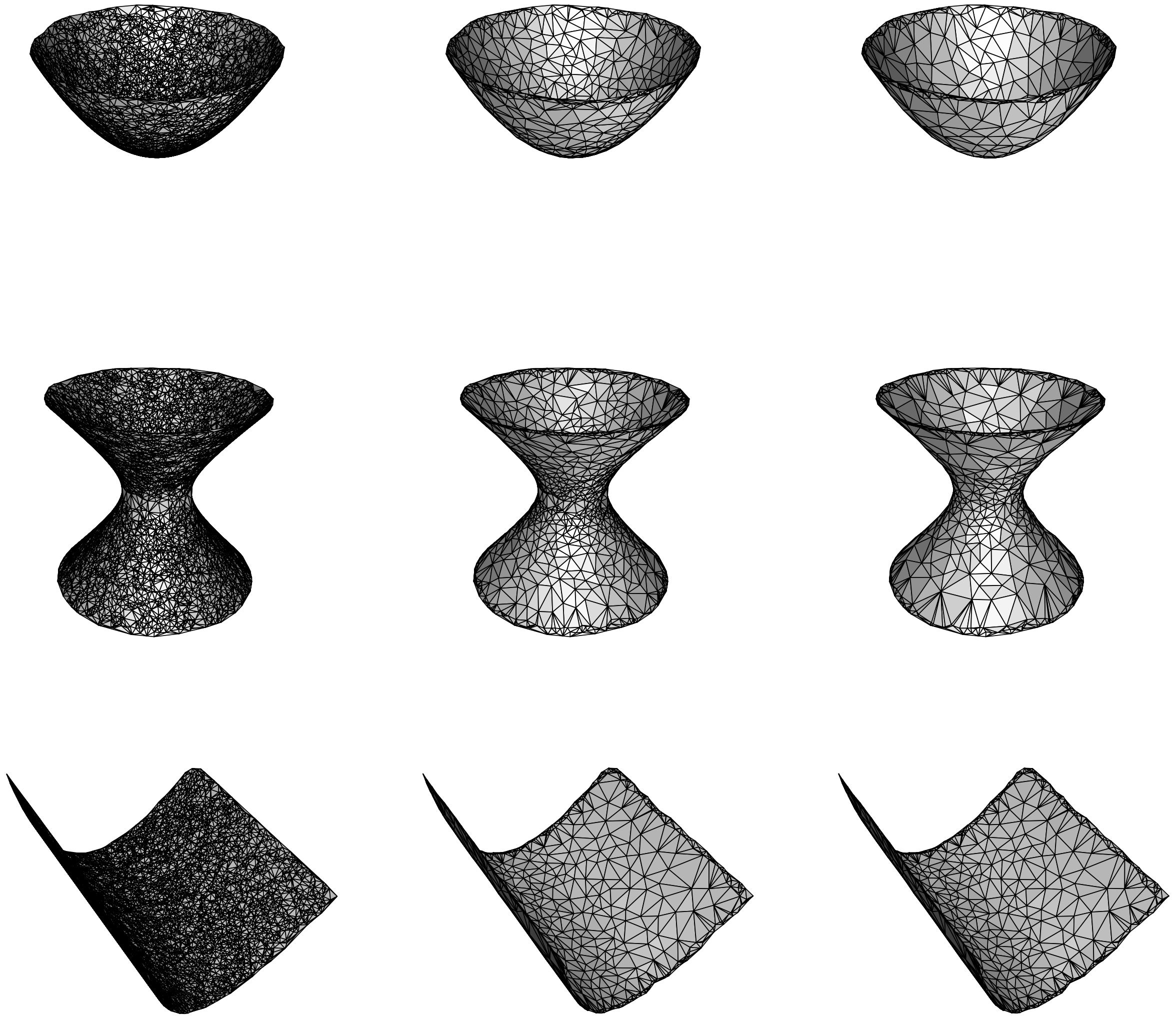

Figure 3. Data sets PARBOL, HYPERBOL, **and** PARCYL. **Leftmost column shows surface meshes without any sample decimation, middle column shows the same after shuffling with $\rho = 0.2, \rho' = 0.13$, rightmost column shows the same after shuffling with $\rho = 0.3$ and $\rho' = 0.2$. Skinny triangles appear only near boundaries as explained in the text.**

point arithmetic. The result of our implementation are shown in Figures 3, 4 and 5. For each data set we compute three surface meshes with the COCONE software [25] for illustrations. The first one is the mesh computed from the original data without decimating it. The other two are computed after decimating the sample by SHUFFLE with the range $\rho = 0.2$ and $\rho' = 0.13$ and the range $\rho = 0.3$ and $\rho' = 0.2$. As expected from theory, the second range achieves more decimations than the first one. The minimum angle and the average absolute deviation Δ of the angles from $60°$ improved with SHUFFLE. Of course, more meshes in the hierarchy can be produced with different ranges of ρ and ρ'.

In our theory we assumed that the sampled surface has no *boundary*. However, we consider some surfaces with boundaries in our experiments. This needs some special teratments for boundaries. In case, boundaries are present, we adopt the heuristic that SHUFFLE does not touch the samples on

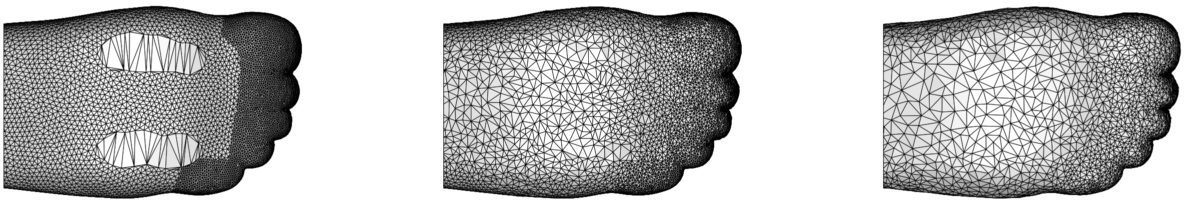

Figure 4. Long and skinny triangles at the bottom of the foot where sample density changes abruptly are smoothed out by SHUFFLE; original (left), decimating with $\rho = 0.2, \rho' = 0.13$ (middle), decimating with $\rho = 0.3, \rho' = 0.2$ (right).

Original data				SHUFFLE, $\rho = 0.2$, $\rho' = 0.13$			SHUFFLE, $\rho = 0.3$, $\rho' = 0.2$		
Name	points	min. ang.	Δ	points	min. ang.	Δ	points	min. ang.	Δ
PARBOL	8000	0.4	23.9	622	21.7	10.3	400	23.1	7.2
HYPERBOL	8000	0.4	23.6	1450	19.2	8.1	871	19.1	8.5
PARCYL	6000	0.3	23.6	1050	22.1	8.5	697	20.6	5.2

Table 1. Experiments with synthetic data

the boundaries. These samples are detected with our COCONE algorithm [10]. As a result we cannot avoid some skinny triangles incident to boundary samples. In measuring minimum angles we did not take into account the triangles incident to boundary vertices and their neighbors.

We consider two categories of data sets for testing, one is synthetically generated from some mathematical surfaces and the other is from real-world applications. We consider three surfaces for synthetic data, paraboloid PARBOL representing elliptic points of positive curvature, hyperboloid HYPERBOL representing hyperbolic points of negative curvature and the parabolic cylinder PARCYL representing parabolic points with zero curvature. We choose these three surfaces since it is known that a quadratic approximation of a smooth surface at any point coincides with one of these three surfaces [21] and the plane. Therefore, the performance of SHUFFLE on these three surfaces provide good indications about its effectiveness on smooth surfaces in the limit when the densities are high. We obtain quite good results with these synthetic data sets as the minimum angle is improved substantially, see Table 1.

Surface meshes for real-world data sets do not have as large minimum angle as the synthetic ones. Nevertheless, SHUFFLE improves both the minimum angle and Δ as Table 2 shows.

For the FOOT data the two sets of parameters achieve 15% and 50% decimations respectively. The originial samples for this data are mostly uniformly distributed over the surface except some sparse patches. As a result Δ is already quite low for the original one though the minimum angle is poor. SHUFFLE improves the minimum angle but does not improve Δ since it is already good. In Figure 4 we zoom in to show how the long and skinny triangles on the bottom of the foot are gradually smoothed out after applying SHUFFLE.

In the TEETH data the number of points increased slightly for the parameter values $\rho = 0.2$ and $\rho' = 0.13$. This is because SHUFFLE needs more points to be inserted than deleted to achieve a better aspect ratio in this case. It achieved only 5% decimation for the second range, but the minimum angle and average angle deviation improved considerably in both cases.

For the ROCKARM data the first parameter range also increased the number of poinst, but improved the quality of the triangles. The second range achieved a 28% decimation with improved angles.

REFERENCES

[1] N. Amenta and M. Bern. Surface reconstruction by Voronoi filtering. *Discr. Comput. Geom.*, **22**, (1999), 481–504.

Original data				SHUFFLE, $\rho = 0.2$, $\rho' = 0.13$			SHUFFLE, $\rho = 0.3$, $\rho' = 0.2$		
Name	points	min ang.	Δ	points	min ang.	Δ	points	min ang.	Δ
FOOT	20021	2.4	11.3	17127	15.9	13.4	9622	15.7	14.1
TEETH	29152	3.8	18.7	35359	14.3	6.2	28384	13.6	5.1
ROCKARM	40177	2.6	16.8	44136	13.1	13.0	29516	12.1	13.2

Table 2. Experiments on real-world data sets

[2] N. Amenta, M. Bern, and D. Eppstein. Optimal point placement for mesh smoothing. *Proc. 8th ACM-SIAM Symposium on Discrete Algorithms*, (1997), 528-537.

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

[4] N. Amenta, S. Choi and R. K. Kolluri. The power crust, unions of balls, and the medial axis transform. *Manuscript*, 2000.

[5] J. D. Boissonnat and F. Cazals. Natural coordinates of points on a surface. *Manuscript*, 2000.

[6] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. *Computing in Euclidean Geometry*, 2nd Ed., World Scientific, 1995, 47–123.

[7] L. P. Chew. Guaranteed-quality mesh generation for curved surfaces. *Proc. 9th Ann. Sympos. Comput. Geom.*, (1993), 274–280.

[8] H.-L. Cheng, T. K. Dey, H. Edelsbrunner and J. Sullivan. Dynamic skin triangulation. *Proc. 12 ACM-SIAM Sympos. Discrete Algorithms*, (2001).

[9] S.-W. Cheng, T. K. Dey, H. Edelsbrunner, M. A. Facello and S.-H. Teng. Sliver exudation. *Proc. 15th Ann. Sympos. Comput. Geom.*, (1999), 1–14.

[10] T. K. Dey and J. Giesen. Detecting undersampling in surface reconstruction. *Proc. 17th Ann. Sympos. Comput. Geom.*, (2001), to appear.

[11] T. K. Dey, H. Edelsbrunner, S. Guha and V. Nekhayev. Topology preserving edge contractions. *Publ. Inst. Math. (Beograd) (N. S.)*, **66**, (1999), 23–45.

[12] H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Trans. Graphics*, **13**, (1994), 43–72.

[13] H. Edelsbrunner and N. Shah. Triangulating topological spaces. *Proc. 10th ACM Sympos. Comput. Geom.*, (1994), 285–292.

[14] P. J. Frey and P.-L. George. *Mesh Generation: application to finite elements*. Hermes Science Publishing, Oxford, 2000.

[15] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. *Proc. SIGGRAPH 97*, (1997), 209–216.

[16] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle. Surface reconstruction from unorganized points. *Proc. SIGGRAPH 92*, (1992), 71-78.

[17] P. Knupp and S. A. Mitchell. Integration of mesh optimization with 3D all-hex mesh generation, Report SAND99-2852, (1999).

[18] X.-Y. Li, S.-H. Teng and A. Üngör. Simultaneous refinement and coarsening for adaptive meshing. *Journal of Engineering with Computers*, **15**, (1999), 280–291.

[19] G. L. Miller, D. Talmor and S.-H. Teng. Optimal good aspect ratio coarsening for unstructured meshes. *Proc. 8th Ann. ACM-SIAM Sympos. Discr. Algorithms*, (1997), 538–547.

[20] S. A. Mitchell and S. A. Vavasis. Quality mesh generation in three dimensions. *Proc. 8th Ann. Sympos. Comput. Geom.*, (1991) 212–221.

[21] B. O'Neill. *Elementary Differential Geometry*, Academic Press, San Diego, (1997).

[22] S. Owen. A survey of unstructured mesh generation technology. `http://www.andrew.cmu.edu/sowen/survey`

[23] J. Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms*, **18**, (1995), 548–585.

[24] J. R. Shewchuk. Tetrahedral mesh generation by Delaunay refinement. *Proc. 14th Ann. Sympos. Comput. Geom.*, (1998), 86–95.

[25] http://www.cis.ohio-state.edu/∼tamaldey/cocone.html

[26] http://www.cgal.org

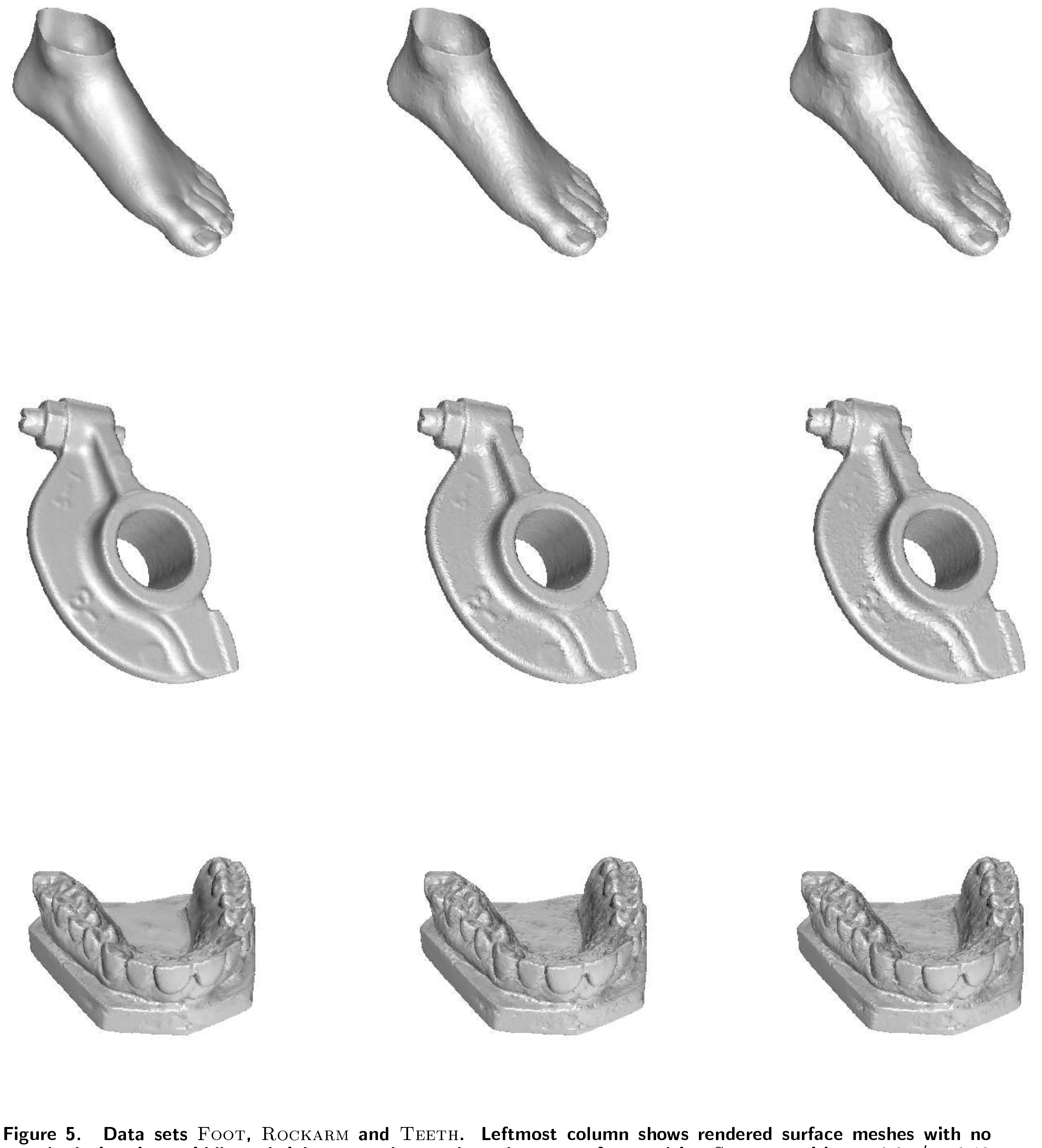

Figure 5. **Data sets** FOOT, ROCKARM **and** TEETH. **Leftmost column shows rendered surface meshes with no sample decimation, middle and rightmost columns show the same after applying** SHUFFLE **with** $\rho = 0.2, \rho' = 0.13$ **and** $\rho = 0.3, \rho' = 0.2$ **respectively. Meshing is done with the** COCONE **software [25].**

APPENDIX

Lemmas used in sample shuffling

Lemma 3.5

Proof. Let $q = \hat{s}$. Step 6 ensures that $||p-q|| \geq \rho\lambda f(s)$. If $||p-q|| \geq \rho\lambda f(p)$ we do not need to prove anything. Otherwise, let $||p-q|| < \rho\lambda f(p)$. In that case $||q-s|| \leq ||p-q|| < \rho\lambda f(p)$, or $||p-s|| < 2\rho\lambda f(p)$. We have $f(s) \geq f(p) - ||p-s|| > f(p)(1-2\rho\lambda)$. Therefore, $||p-q|| \geq \rho\lambda(1-\rho\lambda)f(p) > \frac{\rho\lambda}{2}$ for sufficiently small ρ. □

Lemma 3.6

Proof. Let $R_p \subset S$ denote the set of points that have p as nearest sample in the original sample P. We argue that when any sample $p \in S$ is deleted, there exists some point $q \in Z$ such that q^* is sufficiently near to R_p and q is never deleted in future.

Consider the time when p is deleted. If $\hat{p}$ is inserted, we have $||p-\hat{p}|| \leq \rho h_p$, a condition guaranteed by the inner *while* loop before any sample is deleted. Then, any $x \in R_p$ must have $\hat{p}^*$ within $||x-p|| + ||p-\hat{p}|| + ||p-\hat{p}^*|| = O(\rho)f(p)$ distance (apply Lemma 3.4). Next, assume that $\hat{p}$ is not inserted when p is deleted. In that case $||p-\hat{p}|| \leq \frac{\rho\lambda}{2}f(p)$ by the condition in step 4. Since any q inserted in step 7 is at least $\frac{\rho\lambda}{2}f(p)$ away from p (Lemma 3.5), we must have a point s inserted in step 4 that lies within $\rho\lambda f(p)$ distance of p. Therefore, $||x-s^*|| \leq ||x-p|| + ||p-s|| + ||s-s^*|| = O(\rho)f(p)$ (Lemma 3.4). The lemma follows as $f(p) = \Theta(f(x))$ by Lipschitz property of $f(\cdot)$. □

Lemma 3.7

Proof. Consider the two medial balls going through $p \in S$ with centers on the medial axis. If all points of Q were in S, then these two empty medial balls would have satisfied the lemma. Since Q contains points that are not on S, we cannot assume the medial balls to be empty.

Let B be a medial ball at p with center m and $q \in Z$ be any point inside it. The nearest medial axis point to q^* is closer than m and hence is at most $O(\rho^2)f(q^*)+||p-m|| = O(\rho^2)f(q^*)+O(f(p))$ away from q^*. We get $f(q^*) = O(\rho^2)f(q^*) + O(f(p))$, or $f(q^*) = O(f(p))$.

Consider Figure 6. Any point q only can lie within the shaded region between B and B' where the width of the shaded region is only $O(\rho^2)f(q^*) =$

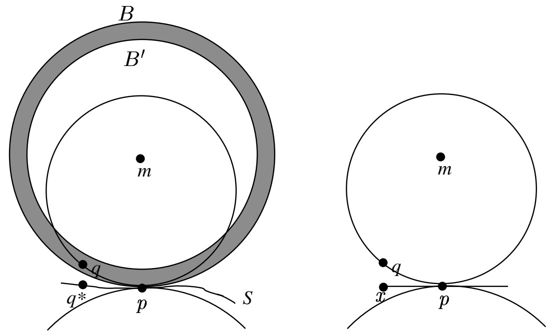

Figure 6. Large tangent balls at p

$O(\rho^2)f(p)$ (Lemma 3.4). To avoid all points of Z we shrink B till its boundary hits the last point, say q. We are interested in a lower bound of the radius of this shrunk ball. The worst case happens when the point q is as close to p as possible, i.e., at a $\Theta(\rho)f(p)$ distance away from p. With this constraint it can be shown that $||q-x|| = O(\rho^2)f(p)$ where x is the foot of the perpendicular dropped from q onto the tangent plane at p. The radius of shrunk B is given by $||p-q||^2/||q-x|| = \Omega(\frac{\rho^2 f(p)^2}{\rho^2 f(p)}) = \Omega(f(p))$ as required by the lemma. □

Lemma 3.8

Proof. Let $x \in V_Z$ be any point in $C_p(\theta, \mathbf{v})$. By the definition of cocone, $\frac{\pi}{2} - \theta - O(\varepsilon) \leq \angle((x-p), \mathbf{n}_p) \leq \frac{\pi}{2} + \theta + O(\varepsilon)$. Let the line of n_p intersect the boundary of V_p at y. Drop the perpendicular pz from p to the line of xy. Observe that either $\angle pxz = \angle((x-p), \mathbf{n}_p) + \angle pyx$, or $\angle pxz = \pi - (\angle((x-p), \mathbf{n}_p) + \angle pyx)$. Since Z^* is an $O(\rho)$-sample and the supporting line of xy intersects S within $C_p(\theta, \mathbf{v})$, we have $||p-z|| = O(\rho)f(p)$. Also, $||p-y|| = \Omega(f(p))$ from Lemma 3.7. It follows that $\angle pyx = O(\rho)$. Therefore, $\angle pxz \geq \frac{\pi}{2} - \theta - O(\varepsilon) - O(\rho) \geq \frac{\pi}{2} - O(\rho)$ and $||p-x|| = ||p-z||/\sin(\angle pxz) = O(\rho f(p))$ proving the lemma. □

Lemma 3.9

Proof. Consider the point set Z that include only the samples in P not deleted so far and the points inserted so far in step 4. According to Lemma 3.6, Z^* is an $O(\rho)$-sample of S. Also, any $s \in Z \setminus S$ satisfies the condition that $||s-s^*|| = O(\rho^2)f(s^*)$ (Lemma 3.4). We can also prove that $||p-s|| = \Omega(\rho)f(p)$ similar to Lemma 3.5. Thus Z satisfies all conditions of Lemma 3.8 and we can conclude that $||p-q|| = O(\rho)f(p)$. Since the current point set is a superset of Z, the cocone at p only gets smaller in the Voronoi diagram of the current point set than in V_Z. Thus, $||p-q|| = O(\rho)f(p)$ when q is inserted. Now we can use the proof of Lemma 3.4 to assert that $||q-q^*|| = O(\rho^2)f(q^*)$ which also gives $f(p) = \Theta(f(q^*))$. The lower bound $||p-q|| = \Omega(\rho)f(q^*)$ follows from the condition in step 7. □

Lemmas used in topological guarantee

Lemma 4.1

Proof. Since $(Q \setminus p) \cup p^*$ satisfies the condition of Lemma 3.7, there are two tangent balls of radii $\Omega(f(p^*))$ going through p^* that are empty of all points in Q except p. Let B and B' be these two balls where B contains p. Shrink B centrally by $||p-p^*|| = O(\rho^2)f(p^*)$ to obtain an empty ball that goes through p. Its radius is $\Omega(f(p^*)) - O(\rho^2)f(p^*) = \Omega(f(p^*))$. In order to obtain the other ball going through p, we translate B' so that it touches p. At this point B' may contain other points of Q that are $\Omega(\rho)f(p^*)$ away from p and we can apply similar logic as in Lemma 3.7 to conclude that B' has $\Omega(f(p^*))$ radius. □

Lemma 4.2

Proof. Consider the two balls B and B' touching at p with radii $\Omega(f(p^*))$ as stated in Lemma 4.1. The centers m and m' of these two balls lie in V_p since they are empty. Also, without loss of generality assume that m and x lie on the opposite sides of S within V_p. It follows that xm intersects S. Consider the triangle pmx. Since the line segment xm intersects S, it has a point within $O(\rho)(p^*)$ distance from p. This means that the perpendicular distance of p from xm is at most $O(\rho)f(p^*)$. Therefore the angle $\angle pmx$ is only $O(\rho)$ since $||p-m|| = \Omega(f(p^*))$ and $||p-y|| = O(\rho)f(p^*)$ where py is the perpendicular dropped from p to pm. Also, since $||p-x|| = \Omega(f(p^*))$ we have $\angle pxm = O(\rho)$. Thus we have $\angle m'px = \angle pmx + \angle pxm = O(\rho)$ proving the claim. □

Lemma 4.4

Proof. Let $x = e \cap S$. The line segment p^*x makes a small angle with the tangent at p^* since the surface S is almost flat locally. In fact this angle is only $O(\rho)$ (Lemma 3.3(a)).

Consider the circle centering p^* with radius $||p^* - x||$ on the plane of the triangle pp^*x. Let the tangent plane at p^* and the parallel plane passing through p intersect the circle at y and z as shown in Figure 7. We are interested in the angle $\angle zpx$. Notice that if x lies above the line of pz, $\angle zpx < \angle yp^*x$ and hence $\angle zpx = O(\rho)$. Otherwise, consider x lying below the line of pz as shown in the figure. In that case $\angle zpx = \angle zpy + \angle ypx$. It follows from Lemma 3.5 that $||p-x|| = \Omega(\rho')f(p^*)$ Also, $||p-p^*|| = O(\rho^2)f(p^*)$. This implies that $||p-y|| > ||p^* - x|| = \Omega(\rho')f(p^*)$. Since the perpendicular from z to py is only $||p-p^*|| = O(\rho^2)f(p^*)$, we have $\angle zpy = O(\rho)$. It is routine to show that $||x-y|| = O(\rho^2)f(p^*)$ since $||p^*-x|| = O(\rho)f(p^*)$ and $\angle yp^*x = O(\rho)$ These conditions imply that $\angle ypx = O(\rho)$. Therefore, $\angle zpx = O(\rho)$. The claim follows since the cocone angle θ is $\frac{\pi}{8}$. □

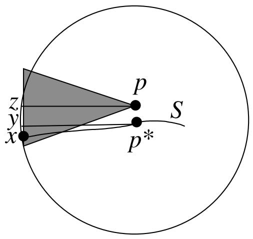

Figure 7. Proof of Lemma 4.4