Size Complexity of Volume Meshes vs. Surface Meshes[*]

Benoît Hudson[†] Gary L. Miller[‡] Todd Phillips[‡] Don Sheehy[‡]

Abstract

Typical volume meshes in three dimensions are designed to conform to an underlying two-dimensional surface mesh, with volume mesh element size growing larger away from the surface. The surface mesh may be uniformly spaced or highly graded, and may have fine resolution due to extrinsic mesh size concerns. When we desire that such a volume mesh have good aspect ratio, we require that some space-filling *scaffold* vertices be inserted off the surface. We analyze the number of scaffold vertices in a setting that encompasses many existing volume meshing algorithms. We show that under simple preconditions, the number of scaffold vertices will be linear in the number of surface vertices.

1 Introduction

Given a surface mesh, many scientific computing and graphics applications will want to produce a volume mesh. Conversely, to build a surface mesh from another description of an input geometry, one might temporarily build a point location structure such as an oct-tree, which is a volume mesh. A natural question arises: can we relate the size of the surface mesh to the size of the volume mesh? A volume mesh will obviously have more vertices than the corresponding surface mesh, but in most settings, the spacing between vertices should grow quickly away from the surface. Since the density of the volume mesh is driven only by the surface, it is intuitive that the surface vertices should dominate in number. Our main result is to show that given a surface mesh in a well-proportioned domain, *the total number of vertices in the volume is linear in the number of vertices on the surface.* We will make this statement specific later as the Scaffold Theorem (Theorem 3.1).

This result has immediate and important ramifications concerning the asymptotic work and space of a large host of existing meshing and surface reconstruction algorithms. For example, in volume meshing, the user may specify a closed surface and ask for its interior to be meshed. Typical algorithms enclose the surface in a *bounding box* that contains the closed surface, incrementally add points until the surface is recovered and the volume mesh has good quality, then strip away the exterior volume vertices (see Figure 1). The surface and interior vertices are then returned to the user. This approach is widespread and is used for many two-, three-, and higher-dimensional meshing algorithms [BEG94, ABE98, She98, CDE+00, ELM+00, MV00, MPW02, Üng04, HMP06, CDR07]. The work and space complexity of these algorithms is output-sensitive and depends on the number of exterior vertices, *even though these vertices are transient.* Our new analysis is the first to control this exterior work. Since we show that the number of transient vertices is bounded by the surface vertices, this for the first time implies that these algorithms run output-sensitively with respect to the true user-desired output.

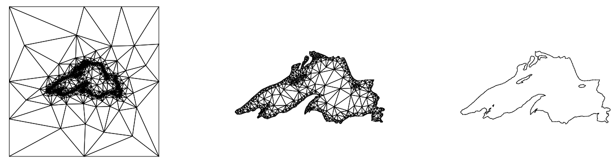

Figure 1: Incremental mesh refinement algorithms first generate a mesh over a bounding box (left), then remove the scaffold vertices and elements (center). Some applications are interested in only the surface mesh (right). The one-dimensional Lake Superior surface mesh shown has 530 surface vertices. The volume mesh shown has 1072 total volume vertices; 258 interior and 284 exterior. We offer the first theoretical analysis of the costs of this scaffolding.

In the rest of this work we make our results precise. A good deal of care is taken to ensure the generality of these results, so that the analysis may be applied to many existing meshing algorithms. Our proofs are in two parts. In the first part, we prove that if a good-quality volume mesh respects a surface, the volume vertices outnumber the surface vertices by only a constant factor. Our definition of respecting a surface is much looser than that of most prior work: the Voronoi cells of the surface vertices must cover the surface, but there is no topological requirement. In addition, our definition of a surface is extremely loose; it need not be manifold, or even connected. Additionally, our surface need not be

[*]This work was supported in part by the National Science Foundation under grants CCF-0635257, CCR-0122581, and CCR-0085982.

[†]Toyota Technological Institute at Chicago

[‡]Carnegie Mellon University

$d-1$ dimensional: for instance, it could be a curve in 3D. Our only requirements are that the surface have a bounded number of connected components, and that each connected component of the surface have diameter within a constant factor of the diameter of the bounding domain.

In the second part, we show how this result relates to standard concepts from mesh refinement and surface reconstruction. In particular, we show that our result proves that a volume mesh of an ϵ-net of a surface is only a constant factor larger than the surface. We also show that many prior quality mesh refinement algorithms are susceptible to our analysis. This implies that they still run in the time (and memory usage) bounds they claim, even when the volume actually meshed is larger than what the user asked to mesh.

Our result is reminiscent of one by Moore [Moo95]. A *balanced* quadtree has neighbouring quadtree cells have size within a constant factor of each other. Given an arbitrary quadtree with m leaves, Moore proved that we can balance the quadtree by splitting only $O(m)$ cells. Indeed, in our proof we very critically use a recent generalization that applies to Voronoi diagrams [MPS08]. What Moore did not discuss is how large m is in relation to some underlying geometric object, such as a point cloud or a surface. This is what the present work establishes.

2 Preliminary Geometric Definitions

In this paper, we assume there exists a surface $\mathcal{S}$ embedded in $\mathbb{R}^d$. For now we allow $\mathcal{S}$ to be any closed subset of space; later, certain requirements will be imposed. In particular, Lemma 4.1 imposes a geometric condition: Let D be the minimum diameter of any connected component of $\mathcal{S}$, where the diameter is the maximum Euclidean distance between two points in the component. We require that the diameter of all other components, and the diameter of $\mathcal{S}$, be in $\Theta(D)$. Around $\mathcal{S}$ there is a compact and connected *domain* Ω with $\mathcal{S} \subset \Omega \subset \mathbb{R}^d$. Typically, Ω will be a box or a hypercube. The diameter of Ω must be in $\Theta(D)$.

Throughout we posit the existence of a set of "constants." By this we mean values that depend only the algorithm, the ambient dimension, and the other constants. That is, the constants cannot depend on the input.

Let Γ_d denote the volume of the unit ball in $\mathbb{R}^d$. For $x \in \mathbb{R}^d$ and $r \in \mathbb{R}$, let $\mathcal{B}(x, r)$ be the open ball centered at x or radius r (whose volume is given by $\Gamma_d\, r^d$).

Suppose we have a set of points (vertices) $M \subset \Omega$. A vertex-set M induces a *local feature size* function $f_M : \Omega \to \mathbb{R}$. At a point $x \in \Omega$, the local feature size is the distance from x to the second-nearest vertex. We frequently use the fact that f_M is 1-Lipschitz: that is, $f_M(x) \leq f_M(y) + |x - y|$ for all x and y in $\mathbb{R}^d$ (this is easily verified by the triangle inequality). At a vertex $v \in M$, the local feature size coincides with the distance to the nearest neighbor, which we denote $NN_M(v)$.

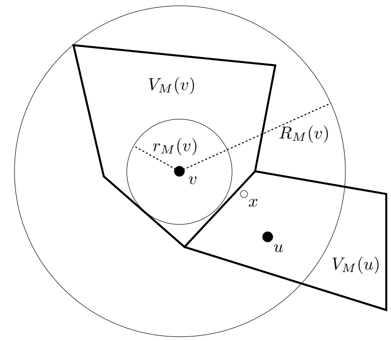

Figure 2: The Voronoi cells of two vertices u and v in a vertex-set M (not pictured). The radii of the inner-ball and outer-ball of v are labeled. The point x is 0.9-medial.

Given the vertex-set M, we denote by $V_M(v)$ the closed Voronoi cell of v: those points in $\mathbb{R}^d$ for which no vertex in M is closer than is v. We identify two natural balls with v: the *inner-ball* $b_M(v)$ is the largest ball centered at v that is contained within $V_M(v)$, while the *outer-ball* $B_M(v)$ is the smallest ball centered at v that contains all of $V_M(v) \cap \Omega$. We denote by $r_M(v)$ and $R_M(v)$ their respective radii (i.e. $B_M(v) = \mathcal{B}(v, R_M(v))$). See Figure 2.

2.1 Well-spaced, Well-paced, and Medial Points

Given a constant τ and a vertex set M such that every vertex v in M has the property $R_M(v) \leq \tau r_M(v)$, we say that v is *τ-well-spaced*. Loosely, this implies that every Voronoi cell is roughly spherical within Ω (it has good aspect ratio), and is roughly centered around its vertex. The right subfigure of Figure 3 shows a set of well-spaced points; contrast this the left subfigure of Figure 4. A set of well-spaced points induces a weighted Delaunay triangulation in which every simplex has good aspect ratio [ELM+00], which is usually what is desired in mesh refinement. Since one can compute a mesh given a well-spaced set of points, in this work we use the two terms interchangeably. Usually we use M to refer to a τ-well-spaced volume mesh, which fills the domain Ω. We use N for those vertices of M that lie exactly on $\mathcal{S}$: N is the surface mesh.

We will make use of a theorem from [MPS08]. First, we introduce some relevant definitions. The boundaries of the Voronoi cells of each vertex in M form the medial axis of M. Miller *et al* [MPS08] generalize this and say that a point x is *θ-medial* with respect to M if it lies near the medial axis, in the sense that $NN_M(x) \geq \theta f_M(x)$. Notice that whenever we add a new point x to the set M, it will decrease the feature size f_M in the vicinity of x. A key observation is that adding a θ-medial point will only decrease the feature-size by a constant fraction.

Given an arbitrary vertex-set N, and an ordered set of vertices $E \equiv \langle v_1, \ldots, v_k \rangle$, we say that E is a θ-well-paced

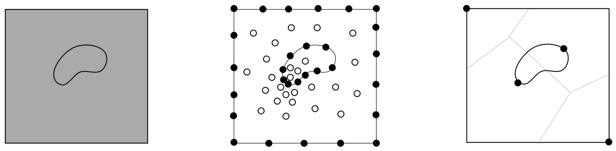

Figure 3: The definitions of Section 3.1 illustrated abstractly. **Left,** a surface $\mathcal{S}$ is composed of *both* black shapes, with the domain Ω shaded. **Center,** vertices form a scaffold mesh M of Ω. The subset of surface vertices N_S are shown in black, with the volume vertices in white. Observe the density of the volume vertices is driven only by the spacing of surfaces vertices. **Right,** a possible seed N_0, containing at least two points from each component. Notice the four points are ρ-well-spaced and have quality Voronoi cells (shown in dashed lines).

extension of N if v_1 is a θ-medial point of N, and each v_i is a θ-medial point of $N \cup \{v_1 \dots v_{i-1}\}$. Informally, the name arises from the fact that the local feature size shrinks only slowly after each insertion.

Well-paced extensions are *not* well-spaced in general, but they have useful similarities to surface meshes. We now state for completeness the Well-Pacing Theorem ([MPS08], Corollary 3).

Theorem 2.1. *(***Well-Pacing Theorem***)*
There is a constant $C_{2.1}$, such that if N is a well-paced extension of a well-spaced set, then there exists a well-spaced superset $M \supset N$, with $|M| \leq C_{2.1}|N|$.

3 Scaffold Theorem

Our main result is the Scaffold Theorem 3.1, showing that given a volume mesh M with underlying surface mesh N, $|M|$ is bounded above by a constant times the size of $|N|$. Informally, we say $|M| \lesssim |N|$. Section 3.1 defines the formal setting in which the Theorem applies.

The main step in the proof is to show that N can be written as a well-paced extension of a well-spaced set. We then apply the Well-Pacing Theorem 2.1 to show that there exists well-spaced superset $|M'| \lesssim |N|$. We show that under reasonable conditions, $|M|$ was only a constant factor worse than the optimal well-spaced superset ($|M| \lesssim |M'|$), and so it follows that $|M| \lesssim |N|$.) We now proceed with a formal proof.

3.1 Definitions: (α, τ)-Scaffold Mesh and ρ-Seed

Suppose we are given a domain Ω as in Section 2. Further suppose we are given a "surface" $\mathcal{S} \subset \Omega$. We require only that $\mathcal{S}$ is a closed subset.

Suppose we have a finite vertex-set $M \subset \Omega$. Define the **surface vertices** $N_S \subset M$ as the minimal subset whose

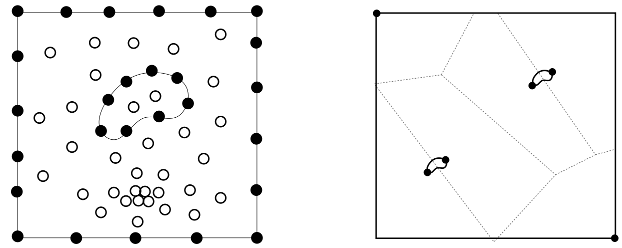

Figure 4: Examples of Non-Scaffold Meshes. **Left,** this volume mesh is not a scaffold mesh, because the sizing is not driven by the surface. The sink in the lower-center could contain arbitrarily many volume vertices. Note how this violates equation (3.4). **Center,** when the surface $\mathcal{S}$ has disproportionately small components, it will be too costly to fill Ω in a way that resolves these small surface features. Note that no seed can exist in this example. An attempted seed is shown, but as the surface components grow relatively small, there is no way to fit two points on each component in a way that is well-spaced.

vertices Voronoi cells cover $\mathcal{S}$, so we have:

$$\mathcal{S} \subset \bigcup_{n \in N_S} V_M(n), \text{ and} \tag{3.1}$$

$$m \in M - N \rightarrow \mathcal{S} \cap V_M(m) = \emptyset \tag{3.2}$$

For $\tau \geq 1$ and $\alpha \in (0, 1)$, we say that M is an (α, τ)-**Scaffold Mesh** for $\mathcal{S}$ in Ω if the following two conditions hold. First, M is τ-well-spaced in Ω:

$$\forall\, m \in M,\ R_M(m) \leq \tau r_M(m) \tag{3.3}$$

Second, $\mathcal{S}$ is responsible for the mesh sizing:

$$\forall\, m \in M,\ r_M(m) \geq \alpha f_{N_S}(m) \tag{3.4}$$

We will require that the volume Ω being filled is somewhat well-proportioned to the underlying surface. We will enforce this by defining the notion of a **ρ-seed** of an embedded graph. Suppose we have a graph G embedded in Ω with vertices N. A ρ-seed N_0 is a ρ-well-spaced subset of N containing at least two vertices for each connected component of G.

3.2 Proof

Theorem 3.1. (Scaffold Theorem) *Suppose M is an (α, τ)-Scaffold Mesh for $\mathcal{S}$ in Ω, and suppose N_S has a ρ-seed N_0, then there exists a constant $C_{3.1}$ depending only on (α, τ, ρ, and dimension d) such that:*

$$|M| \leq C_{3.1}|N|$$

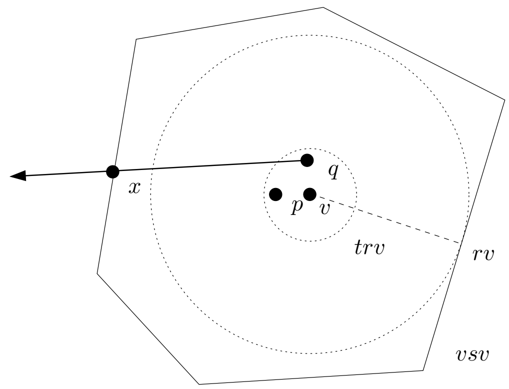

Figure 5: Figure for the proof of Lemma 3.2. If there is still a non-medial uninserted point p, then the original Voronoi cells from M were small near q but large near x, contradicting the assumption that M is well-spaced.

Proof. Let M' be a τ-well-spaced superset of $N_{\mathcal{S}}$ with $|M'| \le C_{3.1}|N|$ as given by Lemma 3.1.

Since M is an (α, τ)-scaffold mesh, standard packing arguments and upper bounds in [Rup95, MV00] guarantee a constant C (depending on α, τ, d) such that $|M| \le C|M'|$. Setting $C_{3.1} = C \cdot C_{3.1}$ proves the theorem.

Lemma 3.1. (Linear-Size Well-Spaced Superset Existence) *Suppose M is a τ-well-spaced set in Ω, with a subset N that has a ρ-seed N_0. Then, there exists a τ-well-spaced superset $M' \supset N$ and a constant $C_{3.1}$ (depending only on τ, ρ, and d) such that:*

$$|M'| \le C_{3.1}|N|$$

Proof. By Lemma 3.2, N is a well-paced extension of the seed N_0, so we simply apply Theorem 2.1 to obtain the existence of $|M|$.

Lemma 3.2. (N admits a well-paced ordering) *Suppose M is a τ-well-spaced set in Ω, with a subset N that has a ρ-seed N_0. Then, $N - N_0$ admits an ordering that is a θ-well-paced extension of N_0, with $\theta = \frac{1}{2+3\tau}$.*

Proof. We construct an ordering by selecting θ-medial vertices greedily, and we prove by contradiction that there is always a θ-medial vertex that can be added to the current set.

Suppose for contradiction that we reach some vertex-set N_i, with $N_0 \subset N_i \subsetneq N$, and there are none of the unadded points are θ-medial with respect to N_i.
Let $p \in N - N_i$ and take $v \in N_i$ such that $p \in V_{N_i}(v)$.
p is not θ-medial, or we would have added it, so it must be relatively close to v in the following sense:

$$|pv| < \theta f_{N_i}(p) \tag{3.5}$$

By the Lipschitz condition on f:

$$\le \theta(|pv| + f_{N_i}(v)) \tag{3.6}$$

And since $v \in N_i$:

$$= \theta|pv| + 2\theta r_{N_i}(v) \tag{3.7}$$

We unravel this as:

$$|pv| < \frac{2\theta}{1-\theta} r_{N_i}(v) \tag{3.8}$$

Consider the Delaunay graph $Del(M)$ on M (given by the dual of the Voronoi diagram), and let G be the subgraph of $Del(M)$ induced by N. Note that G is a subgraph but is not necessarily equal to $Del(N)$. Let e be an edge of G with one endpoint q in $V_{N_i}(v)$ and the other end outside $V_{N_i}(v)$, so that e exits $V_{N_i}(v)$ at some point x. Such an edge must exist, otherwise an entire connected component of G would be contained with in $V_{N_i}(v)$, which would be a contradiction since N_0 is a seed and $S \supset N_0$.

It may be the case that q is equal to p, v, or neither of the two. We consider two cases with only slightly differing arguments, depending on whether $q = v$. First, suppose $q = v$. Because x is on both the boundary of $V_{N_i}(v)$ and a Delaunay edge in M out of v, we have $r_{N_i}(v) \le |xv| \le 2R_M(v)$. Using this along with Eq.3.8, we have:

$$r_M(v) \le \frac{1}{2} NN_M(v) \le \frac{1}{2}|pv| \tag{3.9}$$

$$< \frac{\theta}{1-\theta} r_{N_i}(v) \le \frac{\theta}{1-\theta}|xv| \tag{3.10}$$

$$\le \frac{2\theta}{1-\theta} R_M(v) \le \frac{1}{\tau} R_M(v) \tag{3.11}$$

But this contradicts the assumption that M was τ-well-spaced.

The second case virtually the same except for one more degree of indirection. Now, suppose $q \neq v$. Then $q \notin S$ since $q \in V_{N_i}(v)$, so we have (as before):

$$|qv| < \frac{2\theta}{1-\theta} r_{N_i}(v) \tag{3.12}$$

The triangle inequality yields, $r_{N_i}(v) \le |xv| \le |xq| + |qv|$. Substituting this into Eq. 3.12, we get:

$$|qv| < \frac{2\theta}{1-3\theta}|xq| \tag{3.13}$$

As before, since x on a Delaunay edge of M out of q, we have $|xq| \leq 2R_M(q)$. Using this and Eq. 3.13, we have:

$$r_M(q) \leq \frac{1}{2}NN_M(q) \leq \frac{1}{2}|qv| \tag{3.14}$$

$$< \frac{\theta}{1-3\theta}|xq| \leq \frac{2\theta}{1-3\theta}R_M(q) \tag{3.15}$$

$$= \frac{1}{\tau}R_M(q) \tag{3.16}$$

But again this contradicts the assumption that M was τ-well-spaced. Thus, we can always find a θ-medial point to add, so N is a θ-well-paced extension of N_0.

4 Algorithms

Our result assumes that the surface $\mathcal{S}$, the volume mesh M, the surface mesh N, and the seed N_0 were all given. Ideally, we should not need to know so much, and instead we would have an algorithm to fill in the unknowns. There are many mesh refinement algorithms in the literature that need only know either $\mathcal{S}$ or N. Provided N has a seed, said mesh refinement algorithms will produce an output that matches the requirements of the Scaffold Theorem 3.1: $|M| \in \Theta(|N|)$. The surprising conclusion is that in terms of runtime and output size, when the ambient dimension is bounded, it is asymptotically free to mesh a volume rather than meshing only a surface—again, provided the mesh includes a seed. It is not immediately obvious how to predict whether N has a seed. Therefore, we define a simpler condition that depends only on the geometry of $\mathcal{S}$, and does not depend on N or M:

Definition 4.1. *A ρ-**surface-seed** is a set of points on $\mathcal{S}$ such that every connected component of $\mathcal{S}$ contains at least two two points, and the set is ρ-well-spaced. A surface $\mathcal{S}$ is ρ-**seedable** if it admits a ρ-surface-seed.*

Then we can give a sufficient (though not necessary) condition for ensuring that N has a seed. First, we require that the mesh refinement algorithm refine everywhere sufficiently: no point in Ω is farther than λD from any vertex of N, for some constant λ. This is not much of a restriction: given a mesh that violates the condition, we can add a mere $O(\lambda^{-d})$ points to obey the condition. Second, we require the surface seed vertices not be too close together, from which arises the restriction noted in prior sections that $\mathcal{S}$ must not have any connected component of small diameter. Third, we require that every point $x \in \mathcal{S}$ is in the Voronoi cell of a vertex v on the same connected component of $\mathcal{S}$ as x. This is related to the notion of the *closed ball property* of Edelsbrunner and Shah [ES97], and is obeyed either explicitly or implicitly by most Delaunay mesh refinement algorithms.

Lemma 4.1. *Let $N_0^{\mathcal{S}}$ be a ρ-surface-seed of $\mathcal{S}$. Let λ be an arbitrary constant, and assume every vertex $u \in N_0^{\mathcal{S}}$ has $r_{N_0^{\mathcal{S}}}(u) \geq 3\lambda D$. Let N be the surface vertices of an (α, τ) scaffold mesh of $\mathcal{S}$, and assume that $f_N(x) \leq \lambda D$ for all $x \in \Omega$. Finally, assume that any point $x \in \mathcal{S}$ lies in the Voronoi cell of a vertex on the same connected component. If all these assumptions hold, N contains a $2/\lambda$-seed.*

Proof. For each vertex u of the surface seed, there is a vertex v in N that is closest to u. Let N_0 be the image of this transformation of $N_0^{\mathcal{S}}$. We claim that N_0 is a seed.

For any pair u, and v, we know that $|uv| \leq \lambda D$ by the assumption that f_N is small everywhere, and we know that u and v both lie on the same connected component of $\mathcal{S}$. Consider another vertex u' of the surface seed, and its corresponding v'. We know $|uu'| \geq 3\lambda D$. Then by the triangle inequality, $|vv'| \geq \lambda D$. This proves that N_0 has at least two vertices on every connected component of $\mathcal{S}$.

It remains to prove that N_0 is well-spaced. We know that u had $r_{N_0^{\mathcal{S}}}(u) \leq R_{N_0^{\mathcal{S}}}(u)$. When both u and its nearest neighbour u' are mapped, their images may move closer together, but they remain at least $|vv'| \geq \lambda D$ apart as proved above: $r_{N_0}(v) \geq \lambda D/2$. While we could prove a tighter bound, it suffices to note that $R_{N_0}(v) \leq D$ to show that N_0 is $2/\lambda$-well-spaced.

In the remainder of this section, we assume the first two conditions of Lemma 4.1 apply, and we will argue that the third condition is maintained by the algorithms we analyze.

4.1 Meshing a surface sample: The simplest application is to take as input a set of points N that all lie on a manifold surface (for example, the famous Stanford Bunny model), and construct from it the volume mesh M. This is a useful endeavor if we are to animate the model. Here, we need not assume that $\mathcal{S}$ is known. To generate the volume mesh, we use a Voronoi (or Delaunay) refinement algorithm. The volume mesher first wraps the points of N into an appropriate bounding box, of diameter only a constant factor larger than the diameter of N. It initializes M with N, then finds a vertex v with $R_M(v) \geq \tau r_M(v)$, and identifies some point p that is in the Voronoi cell of v, but far from it: $|pv| \leq |pu|$ for all $u \in M$, but $|pv| \geq \tau r_M(v)$. The algorithm then adds p to M, and continues this process until M is τ-well-spaced. A large number of algorithms implement this process (*e.g.* [Rup95, She98, HPU05, HMP06]).

Our theorem requires that the surface is covered by the Voronoi cells of just the surface vertices—that is, no point of $\mathcal{S}$ lies in the Voronoi cell of a vertex in $M \backslash N$. Under certain assumptions on N, we can prove this holds. We require that there be some ϵ such that for all x on $\mathcal{S}$, there is a vertex $v \in N$ such that $|vx| \leq \epsilon$; but for all $u \in N$, all other vertices $v \in N$ lie at distance $|uv| \geq \epsilon/2$. In other words, N is an ϵ-net of $\mathcal{S}$.

Lemma 4.2. *Consider a point $x \in \mathcal{S}$ whose nearest neighbor in N is v. If the volume mesh M is computed with $\tau > 4$, then x remains in the Voronoi cell $V_M(v)$.*

Proof. For any vertex u created during refinement, there is some u' that created u: when u was inserted, its nearest neighbor was u', and $|uu'| \geq \tau r_M(u')$. In other words, created vertices have nearest neighbor larger than the distance between the closest pair of points in N. The closest pair must be at least $\epsilon/2$ from each other, by assumption, so any $u \in M \backslash N$ has $r_M(u) \geq \tau\epsilon/4$. Return now to consider x. For a contradiction, we assume that the nearest neighbor of x in N was v, but its nearest neighbor in M is a created vertex u. Then $|ux| < |vx|$. Given that $|vx| \leq \epsilon$, we know that $|uv| \leq 2\epsilon$. By definition, $r_M(u)$ is half the distance from u to its nearest neighbor: $r_M(u) \leq |uv|/2 \leq \epsilon$. Remembering the lower bound on $r_M(u)$, we conclude that $\epsilon \geq \tau\epsilon/4$, or equivalently, $\tau \leq 4$, a contradiction since we chose $\tau > 4$.

As a corollary, this means that every point $x \in \mathcal{S}$ is in the Voronoi cell of some vertex in N, and therefore Theorem 3.1 holds. Then $|M| \in O(|N|)$, assuming N is an ϵ-net of $\mathcal{S}$, and that $\tau > 4$. But N is input, so $n = |N|$: the volume mesh contains a number of vertices only linear in the size of the input! We can relax the requirement on τ by remembering that a τ-well-spaced mesh and a τ'-well-spaced mesh have size within a constant factor of each other, where the constant is a function of τ, τ'. This lets us conclude:

Corollary 4.1. *A τ-well-spaced mesh of an ϵ-net has size $O(n)$ for any τ and ϵ.*

4.2 Meshing a surface: In mesh refinement for engineering and scientific applications, the input is typically specified as a piecewise linear complex or a piecewise smooth complex, made up of a collection of vertices, segments or curves, and polygons or smooth surfaces (and so on, in higher dimension). As in the prior subsection, we assume the algorithm first places a box around the input complex, then iteratively inserts vertices. In the face of linear or smooth features, this requires greater care than before although the details are nearly irrelevant to our results here. The mesher continues adding vertices until two conditions are met: that the vertices are well-spaced, and that the Delaunay triangulation "respects" the input complex. In the case of piecewise linear complexes, we say a triangulation respects it if each linear facet appears as the union of a set of Delaunay simplices [Rup95, She98]. In other words, the Voronoi diagram of surface vertices covers the input. There is an analogous condition for smooth complexes [CDR07, RY07].

The analysis of algorithms that mesh complexes typically rely on a notion of a local feature size defined by the surface $\mathcal{S}$ rather than by a set of points. To reduce confusion, we use $\text{lfs}(x)$ to denote this local feature size function. For our purposes, we require that the local feature size be defined on $\mathcal{S}$, and extended to the entire domain via the minimum 1-Lipschitz function: at $x \in \Omega \backslash \mathcal{S}$, $\text{lfs}(x) \equiv \min_{y \in \mathcal{S}} \text{lfs}(y) + |xy|$. This is within a factor of three of Ruppert's more traditional local feature size function defined on linear complexes, but extends more easily to smooth complexes. Most mesh refinement algorithms arising from the computational geometry community guarantee that vertices are not too closely packed: the algorithm defines a constant γ^- such that at every $v \in M$, $r_M(v) \geq \gamma^- \text{lfs}(v)$.

Lemma 4.3. *For all $x \in \Omega$, $f_N(x) \geq \frac{\gamma^-}{1+\gamma^-} \text{lfs}(x)$.*

Proof. For the lower bound on f_N, consider a point x in the domain. It lies in the Voronoi cell of some vertex $v \in M$. Since local feature size is 1-Lipschitz, $\text{lfs}(x) \leq \text{lfs}(v) + |vx|$. By the assumption on the algorithm, $r_M(v) \geq \gamma^- \text{lfs}(v)$. We also know that the second-nearest vertex to x is at least as far as $\max(r_M(v), |vx|)$. Thus we know $\text{lfs}(x) \leq (1 + 1/\gamma^-) f_M(x)$. Finally, removing vertices can only increase f: $f_M(x) \leq f_N(x)$, which proves that $f_N(x) \geq \frac{\gamma^-}{1+\gamma^-} \text{lfs}(x)$.

This shows that M, the mesh output by a typical Delaunay mesh refinement algorithm, is a $(\frac{\gamma^-}{1+\gamma^-}, \tau)$-scaffold mesh for $\mathcal{S}$.

Corollary 4.2. *When presented with an input piecewise smooth or piecewise linear complex for which there exists a seed, a quality mesh refinement algorithm that outputs a mesh of optimal size creates a volume mesh M and a surface mesh N, with $|M| \in O(|N|)$.*

5 Conclusions

Accounting for scaffolding costs is a pressing question in the timing and output-size analysis of many mesh generation algorithms that are used in practice. The Scaffold Theorem shows that these costs are not dominant, as has so often been assumed without proof in prior work. This analysis is made applicable to many algorithms by abstracting the meshing problem to that of simply generating a minimal well-spaced superset of a vertex-set. This ignores many of the topological and geometric intricacies that make meshing algorithms difficult to analyze, while still preserving enough distribution information about the vertices to make meaningful statements on mesh-size.

Reflecting on the analysis, the surface vertices are paramount and the underlying surface itself plays only a small role in controlling the size of the volume mesh, It is then theoretically of interest to simply consider the size of a minimal well-spaced superset M of a vertex-set $N \subset \Omega$.

It is well-established that:

$$|M| \in \Theta\left(\int_{\Omega} \frac{1}{f_N^d}\right)$$

A worst case upper bound on this integral is $O(|N| \log \Delta)$, where Δ is the spread of the domain; the ratio of the diameter of Ω to the closest pair in N. In general, this bears no combinatorial relationship to $|N|$. The Scaffold Theorem provides sufficient conditions (that are highly relevant in practice) for a setting wherein $|M|$ is linear in $|N|$. But these conditions are nowhere near necessary. It is an interesting question whether there exist simple necessary and sufficient conditions that will combinatorially bound $|M|$ when N is given arbitrarily.

References

[ABE98] Nina Amenta, Marshall Bern, and David Eppstein. The crust and the β-skeleton: Combinatorial curve reconstruction. *Graphical models and image processing: GMIP*, 60(2), 1998.

[BEG94] Marshall Bern, David Eppstein, and John R. Gilbert. Provably Good Mesh Generation. *Journal of Computer and System Sciences*, 48(3):384–409, June 1994.

[CDE+00] Siu-Wing Cheng, Tamal Krishna Dey, Herbert Edelsbrunner, Michael A. Facello, and Shang-Hua Teng. Sliver Exudation. *Journal of the ACM*, 47(5):883–904, September 2000.

[CDR07] Siu-Wing Cheng, Tamal K. Dey, and Edgar A. Ramos. Delaunay refinement for piecewise smooth complexes. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1096–1105, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[ELM+00] Herbert Edelsbrunner, Xiang-Yang Li, Gary L. Miller, Andreas Stathopoulos, Dafna Talmor, Shang-Hua Teng, Alper Üngör, and Noel Walkington. Smoothing and cleaning up slivers. In *STOC*, pages 273–277, 2000.

[ES97] Herbert Edelsbrunner and Nimish R. Shah. Triangulating topological spaces. *IJCGA*, 7:365–378, 1997.

[HMP06] Benoît Hudson, Gary Miller, and Todd Phillips. Sparse Voronoi Refinement. In *Proceedings of the 15th International Meshing Roundtable*, pages 339–356, Birmingham, Alabama, 2006. Long version available as Carnegie Mellon University Technical Report CMU-CS-06-132.

[HPU05] Sariel Har-Peled and Alper Üngör. A Time-Optimal Delaunay Refinement Algorithm in Two Dimensions. In *Symposium on Computational Geometry*, 2005.

[Moo95] Doug Moore. The cost of balancing generalized quadtrees. In *SMA '95: Proceedings of the Third Symposium on Solid Modeling and Applications*, pages 305–312, 1995.

[MPS08] Gary L. Miller, Todd Phillips, and Donald Sheehy. Linear-size meshes. In *Canadian Conference on Computational Geometry*, 2008. To appear.

[MPW02] Gary L. Miller, Steven E. Pav, and Noel J. Walkington. Fully Incremental 3D Delaunay Refinement Mesh Generation. In *Eleventh International Meshing Roundtable*, pages 75–86, Ithaca, New York, September 2002. Sandia National Laboratories.

[MV00] Scott A. Mitchell and Stephen A. Vavasis. Quality mesh generation in higher dimensions. *SIAM J. Comput.*, 29(4):1334–1370 (electronic), 2000.

[Rup95] Jim Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms*, 18(3):548–585, 1995. Fourth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA) (Austin, TX, 1993).

[RY07] Laurent Rineau and Mariette Yvinec. Meshing 3D domains bounded by piecewise smooth surfaces. In *16th International Meshing Roundtable*, pages 442–460, 2007.

[She98] Jonathan Richard Shewchuk. Tetrahedral Mesh Generation by Delaunay Refinement. In *Proceedings of the Fourteenth Annual Symposium on Computational Geometry*, pages 86–95, Minneapolis, Minnesota, June 1998. Association for Computing Machinery.

[Üng04] Alper Üngör. Off-centers: A new type of steiner points for computing size-optimal guaranteed-quality delaunay triangulations. In *Proceedings of LATIN*, 2004.