NEWEST VERTEX BISECTION OVER GENERAL TRIANGULATIONS

MICHAEL HOLST, MARTIN LICHT, AND ZHAO LYU

Abstract. We give an accessible exposition and analysis to algorithmic mesh refinement via newest vertex bisection for triangular meshes. We prove a purely combinatorial amortized complexity estimate whose constant depends only on the mesh topology but not on geometric quantities. Furthermore, we extend newest vertex bisection to two-dimensional triangulations of arbitrary topology. This includes but is not restricted to triangulations of two-dimensional embedded manifolds.

1. Introduction

The algorithmic local refinement of triangulations is an important topic of scientific computing and poses interesting research problems in computational geometry and combinatorics. A variety of approaches, which differ in their refinement strategies, have been published in the literature. A popular class of mesh refinement algorithms utilizes *newest vertex bisection*: triangles are bisected along the edge opposite to the most recently created (that is, *newest*) vertex.

This exposition gives a thorough combinatorial description of newest vertex bisection and its amortized complexity analysis in two dimensions. Our main result is an amortized complexity analysis of newest vertex bisection that is purely combinatorial and does not depend on geometric quantities. We describe and analyze newest vertex bisection for a class of triangulations larger than those in prior works. In particular, we cover embedded surfaces of arbitrary topology and singular surfaces that appear, for example, in numerical simulations over surfaces.

We now give an outline of local mesh refinement and newest vertex bisection to establish the background of this research. The general problem setting of local mesh refinement is the following: given an initial triangular mesh, a user marks triangles for refinement, and the refinement algorithm then *refines*, that is, subdivides those marked triangles and possibly other triangles too. This procedure of marking and refining is repeated over and over again, producing a sequence of progressively refined meshes. In this context the user might be, for example, a numerical simulation code. The mesh refinement algorithm is an *online algorithm* in

2010 *Mathematics Subject Classification.* 05E45,68U05,65N50.

Key words and phrases. amortized complexity, mesh refinement, newest vertex bisectin, online algorithm.

The authors would like to thank the Isaac Newton Institute for Mathematical Sciences, Cambridge, for support and hospitality during the programme "Geometry, compatibility and structure preservation in computational differential equations" where work on this paper was undertaken. This work was supported by EPSRC grant no EP/K032208/1. MH was supported in part by NSF DMS/FRG Award 1262982 and NSF DMS/CM Award 1620366. ML was supported in part by NSF DMS/RTG Award 1345013 and DMS/CM Award 1262982.

the terminology of theoretical computer science since the user reveals the markings of triangles only successively.

The meshes produced by the refinement algorithm need to satisfy two requirements, one combinatorial and the other geometric; see Figure 1. On the one hand, we require the meshes to satisfy the combinatorial condition of *consistency*. This simply means that triangles are either disjoint or meet at a common vertex or edge. Many data structures in implementations can only represent consistent meshes. On the other hand, we require geometric *stability* for the sequence of meshes, which loosely means that the triangles produced retain good shape. Mathematically, their angles are bounded from below. Having both requirements satisfied at the same time is non-trivial and has inspired a variety of different refinement algorithms in the literature.

We focus on newest vertex bisection (NVB) in this article. Here, we divide a triangle along the line from its newest vertex to the midpoint of the opposite edge, which is also called the *refinement edge* of that triangle. That midpoint is then the most recently created, *newest*, vertex of the two new triangles, and hence the newest vertex bisection can be repeated. It is easy to prove geometric stability if all refinements are newest vertex bisections. To ensure that we only generate consistent meshes, we must further restrict ourselves to *compatible bisections*: the simultaneous bisection of all triangles sharing a common edge that is the refinement edge of all those triangles.

We will describe how for every selection of triangles to be refined one can recursively construct a finite sequence of compatible bisections whose successive execution will refine the selected triangles. The bisection of any triangle may depend on the prior bisection of its neighboring triangles. Our mesh refinement algorithm processes those dependencies recursively.

The refinement of any selection of triangles invokes additional refinements in our recursive algorithm, and an amortized complexity estimate asserts that the additional work stays within feasible bounds: over repeated calls to the mesh refinement algorithm, the total number of triangles bisected is linearly bounded in the numbers

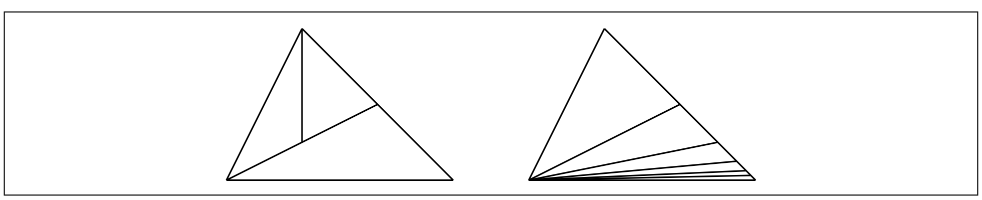


FIGURE 1. Left: inconsistent refinement of a triangle. Right: unstable refinement of a triangle.

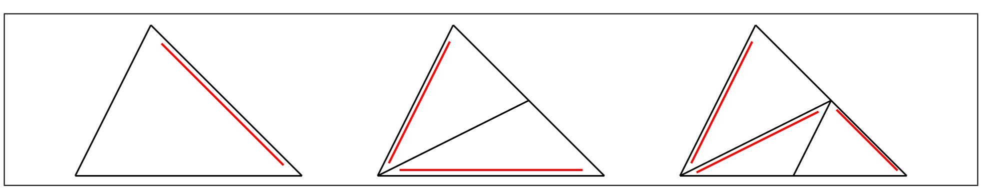


FIGURE 2. Illustration of a sequence of newest vertex bisections. Refinement edges indicated with a bar.

of triangles ever selected by the user. The constant in the upper bound depends on a few parameters of the initial mesh. Such an amortized complexity estimate was proven first for the recursive newest vertex bisection of Binev, Dahmen and DeVore [6] in two dimensions and later by Stevenson [36] in arbitrary dimension. Different versions of the basic proof technique have appeared in the literature (see Nochetto [31] or Karkulik, Pavlicek and Preatorius [24]).

The upper bounds in the amortized complexity estimates depend on geometric properties of the mesh. We elaborate how the upper bound grows with the ratio of the largest diameter and the smallest volume of the triangles in the initial triangulation. Thus the upper bound will deteriorate if the initial mesh is highly non-uniform. But the presence of geometric quantities in the complexity estimate is counterintuitive in the first place, since newest vertex bisection is a purely combinatorial algorithm that does not see the geometry. We "combinatorialize" the basic proof technique in the literature and state a purely combinatorial constant in our complexity estimate. The constant depends only on the maximum number of triangles adjacent to any vertex. Such a combinatorial estimate is naturally expected, since newest vertex bisection is defined in purely combinatorial language, and any dependence on geometric properties must therefore be an artifact of the proof.

This exposition provides a technical reference for newest vertex bisection over triangulated surfaces. Most of the sources on mesh refinement focuses on triangulations of domains. Our exposition analyzes newest vertex bisection for arbitrary embedded triangulations of dimension two without any constraints on the topology. This includes triangulations of embedded surfaces, as well as "singular" surfaces that are not manifolds anymore.

Localized mesh refinement is a fundamental for the feasible numerical solution of partial differential equations and its amortized complexity analysis enters the complexity analysis of adaptive finite element methods. Surface finite element methods have seen a surge of research activity in recent years [18, 14, 33, 23, 20, 19, 9, 10, 11], and numerical simulations over singular surfaces have recently attracted attention [21, 32]. Our analysis enables analogous complexity bounds for adaptive finite element methods over surfaces [22, 17, 8, 12, 15, 7]. When triangular surfaces approximate an embedded two-dimensional manifold, then the purely combinatorial refinement is often followed by additional geometric transformations, such as moving newly created vertices onto the true surface [16, 28]. In that regard, a geometry-independent purely combinatorial complexity estimate is clearly advantageous.

We finish this introduction with a brief overview of bisection algorithms. Even though mesh refinement is a topic of computational geometry and combinatorics, many contributions have come from scholars of numerical analysis. The idea of newest vertex bisection has appeared in two classes of algorithms, called *iterative* and *recursive* [25, 30], which differ in how the global mesh consistency is preserved. The iterative algorithms perform bisections that temporarily lead to inconsistent meshes; additional *closure* bisections repair these inconsistencies but may lead to new inconsistencies themselves. Closure bisections are performed until no new inconsistency has been produced. Such iterative refinement algorithms include the methods proposed by Bänsch [3], Liu and Joe [26], and Arnold, Mukherjee and

Pouly [1]. We remark that Rivara [34, 35] describes a iterative longest edge bisection. By contrast, recursive algorithms always perform newest vertex bisections that retain consistency; if a triangle is to be bisected, the algorithm constructs a sequence of bisections which always preserve the mesh consistency and eventually lead to bisection of the original triangle. Recursive algorithms do not require data structures to handle temporary intermediate inconsistent meshes, which makes them much easier to implement in practice. Recursive refinement algorithms have been studied by Mitchell [29], Kossaczky [25], Traxler [37], Maubach [27], and Stevenson [36], and they are also the subject of this article. The *coarsening* of triangulations has been studied by Bartels and Schreier [4] and by Long and Zhang [13].

It is our explicit hope that our exposition encourages research in local mesh refinement through techniques from theoretical computer science and combinatorics. We would like to highlight what we believe are two particularly interesting possibilities to extend upon the results in this article by combinatorial techniques.

On the one hand, the assignment of initial refinement edges, which is equivalent to finding a perfect matching in a particular class of graphs, is algorithmically and mathematically interesting in its own right. We are aware of the sequential linear-time algorithm by Biebl, Bose, Demaine, and Lubiw [5]. To our best knowledge, not much is known about the parallel complexity of this problem. On the other hand, we believe that there is considerable room for improving the constant in our amortized analysis. We mention work by Atalay and Mount [2] that bounds the number of newest vertex bisections necessary to rebuild consistency for a non-consistent triangulation that has resulted from several newest vertex bisections; their result translates to a purely combinatorial complexity estimate when the only triangles marked are children of previously marked triangles. We allow a more generally marking of triangles but our computational experiments indicate that our upper bound is not sharp.

The remainder of this paper is structured as follows. In Section 2 we review the geometric aspects of newest vertex bisection over a single triangle. In Section 3 we introduce the combinatorial structures to be used for the newest vertex bisection algorithm over entire triangulations. We develop the global refinement procedure in Section 4 and review different forms of the global newest vertex bisection algorithms in Section 5. We prove the asymptotic complexity estimate in Section 6. Finally, algorithms for the initial assignment of refinement edges are reviewed in Section 7.

2. Geometry and Similarity Classes

In this section we discuss geometric quantities associated with triangles and describe newest vertex bisection repeatedly applied to a single triangle. We prove geometric stability and bound the number of similarity classes.

A triangle T in $\mathbb{R}^d$ is the convex closure of three affinely independent points. We write $\operatorname{diam}(T)$ for the diameter and $\operatorname{meas}(T)$ for the two-dimensional measure of T. For any edge e of the triangle T, we write $|e|$ for the length of that edge. The *shape measure* $\mu(T)$ is the quantity

$$\mu(T) := \frac{\operatorname{diam}(T)^2}{\operatorname{meas}(T)}. \tag{1}$$

The shape measure quantifies how far a triangle is from being degenerate. For example, a flat triangle has small measure but large diameter and hence it has a large shape measure. The shape measure $\mu(T)$ quantifies in how far the edges of T have comparable lengths. For any edge e of the triangle T we have

$$|e| \leq \operatorname{diam}(T) \leq \mu(T)|e|, \tag{2}$$

as follows from $|e| \operatorname{diam}(T) \geq 2 \operatorname{meas}(T) \geq \operatorname{diam}(T)^2/\mu(T)$.

Henceforth, for every triangle we choose one of its three edges and call that edge the *refinement edge* of that triangle. The vertex opposite to the refinement edge is called the *refinement vertex* of the triangle. Letting $T = [v_0, v_1, v_2]$ be a triangle with refinement edge $[v_1, v_2]$ and letting $v_N = \frac{1}{2}(v_1 + v_2)$ denote the midpoint of T's refinement edge, the *children* of T are the triangle $T^- = [v_0, v_1, v_N]$ with refinement edge $[v_0, v_1]$ and the triangle $T^+ = [v_0, v_2, v_N]$ with refinement edge $[v_0, v_2]$. Conversely we call T the *parent* of T^- and T^+. Note that the refinement edges of the children are the respective edges opposite to v_N. We say that a triangle S is a *descendant* of T if either $S = T$, or S is a child of T, or S is a child of a descendant of T. Note that the refinement vertex of any triangle created is always the newest vertex, which is why we call this *newest vertex bisection* (NVB).

We introduce the notion of *level* whenever the original triangle T is understood: the original triangle T has level $\ell(T) = 0$, and whenever a triangle S has level $\ell(S) \in \mathbb{N}_0$, then its children S^- and S^+ have level $\ell(S^-) = \ell(S^+) = \ell(S) + 1$.

Recall that a *similarity transformation* is any combination of translations, scalings, and orthogonal transformations. We call two triangles T and T' with their respective choice of refinement edge are *similar* if they can be mapped to each other by a similarity transformation that maps the refinement edge of T onto the refinement edge of T'. This defines an equivalence classes and we call the corresponding equivalence classes *similarity classes*. When two triangles T and T' are similar, then $\mu(T) = \mu(T')$.

For similar triangles T and T' we have $\mu(T) = \mu(T')$. This follows easily by considering the effects of translations, scalings, and orthogonal transformations on diameters and measures of triangles.

Theorem 2.1. *The descendants of a triangle T belong to no more than four similarity classes. If S is a descendant of T, then $\mu(S) \leq 4\mu(T)$*

Proof. Let T be a triangle with refinement edge. We consider the *standard triangle* $\widehat{T} \subset \mathbb{R}^2$ whose vertices are the origin and the two standard unit vectors and whose refinement edge is the edge opposite the origin. We fix an affine mapping ϕ which maps $\widehat{T}$ onto T and which maps the refinement edge onto the refinement edge. We observe that ϕ maps the descendents of $\widehat{T}$ onto the descendents of T in a one-to-one manner.

The descendants of $\widehat{T}$ of generation one, two and three are depicted in Figure 3. We divide those descendants into four types as indicated in that figure. Within each type, the triangles are not only similar but can be mapped onto each other using only translations and positive scalings. Consequently, the triangles of the same type are mapped under ϕ onto the same similarity class. We conclude that the descendants of T fall into only four similarity classes.

Since bisection halves the triangle measure, the descendants of $\widehat{T}$ of generation zero, one, and two have measure at least $\operatorname{meas}(\widehat{T})/4$. We conclude that analogously the descendants of T of generation zero, one, and two have measure at least $\operatorname{meas}(T)/4$. Since the descendants of T of generation up to two already are instances for all similarity classes of the descendants of T, and similar triangles have the same shape measure, the desired estimate on the shape measures of T's descendants follows. □

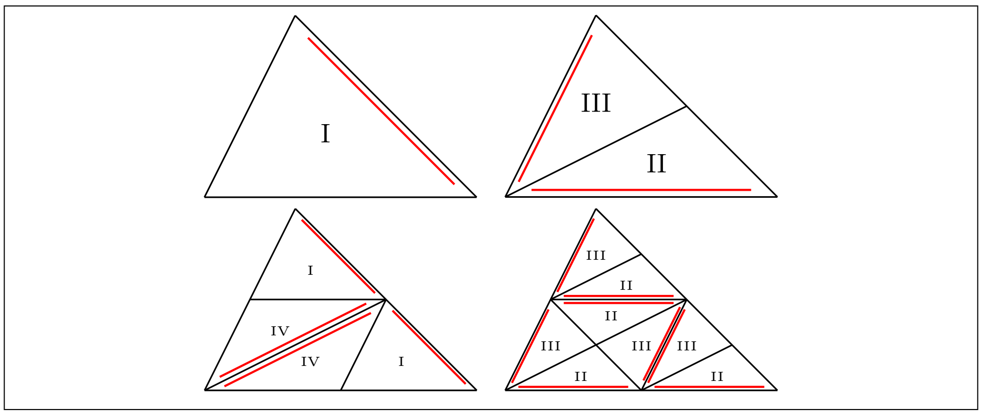


FIGURE 3. Descendants of the standard triangle after three refinement steps when the refinement edge is opposite the origin. Affine transformations preserve the indicated similarity classes.

Based on the preceding theorem, we can relate the diameters and measures of descending triangles to the diameter and measure of the original triangle as in the following lemma.

Lemma 2.2. *Suppose that T is a descendant of the triangle T_0. Then*

$$\operatorname{meas}(T) = 2^{-\ell(T)} \operatorname{meas}(T_0). \tag{3}$$

Furthermore,

$$2 \cdot 2^{-\ell(T)} \mu(T_0)^{-1} \operatorname{diam}(T_0)^2 \leq \operatorname{diam}(T)^2 \leq 4 \cdot 2^{-\ell(T)} \operatorname{diam}(T_0)^2. \tag{4}$$

Proof. The first identity follows by an induction argument, using that the children triangles have the same height as their parent triangle but exactly half of their parent's base length. For the second identity, we use Theorem 2.1 and calculate

$$\begin{aligned} \operatorname{diam}(T)^2 = \mu(T) \operatorname{meas}(T) &\leq 4\mu(T_0) \operatorname{meas}(T) \\ &= 4\mu(T_0) 2^{-\ell(T)} \operatorname{meas}(T_0) = 4 \cdot 2^{-\ell(T)} \operatorname{diam}(T_0)^2. \end{aligned}$$

On the other hand, since $\mu(T) \geq 2$, we get

$$\operatorname{diam}(T)^2 \geq 2 \operatorname{meas}(T) = 2 \cdot 2^{-\ell(T)} \operatorname{meas}(T_0) = 2 \cdot 2^{-\ell(T)} \mu(T_0)^{-1} \operatorname{diam}(T_0)^2.$$

The proof is complete. □

Remark 2.3. *The estimate for the descendant's shape measure in Theorem 2.1 is optimal, as can be seen from newest vertex bisection of an isosceles triangle whose refinement edge is a narrow base.*

3. Triangulations and Compatible Divisions

In this section we formally define triangulations and several combinatorial structures that help us understand the recursive mesh refinement algorithm that we introduce later in this article.

A *triangulation* is a collection $\mathcal{T}$ of triangles in $\mathbb{R}^d$ such that for every two distinct triangles $T, T' \in \mathcal{T}$ the intersection $T \cap T'$ is either empty, a common edge of T and T', or a common vertex of T and T'.

We let $\mathcal{E} = \mathcal{E}(\mathcal{T})$ be the set of edges of the triangles in $\mathcal{T}$ and let $\mathcal{V} = \mathcal{V}(\mathcal{T})$ denote the set of vertices of the triangles in $\mathcal{T}$. We call an edge $e \in \mathcal{E}$ a *border edge* if it is shared by only one single triangle $T \in \mathcal{T}$; otherwise it is shared by more than one triangle and we call it an *interior edge* then. Likewise, we call $v \in \mathcal{V}$ a *border vertex* if it is contained in a border edge and we call it *interior vertex* otherwise.

We emphasize that every triangle is assumed with a choice of refinement edge. We write $\mathcal{R}(T) \in \mathcal{E}$ for the refinement edge of a triangle $T \in \mathcal{T}$ and $\mathcal{R}(\mathcal{T}) \subseteq \mathcal{E}$ for the set of refinement edges of all triangles in $\mathcal{T}$.

We call a triangulation $\mathcal{T}$ *manifold-like* if every edge $e \in \mathcal{E}$ is shared by at most two different triangles. Obviously, that means precisely that all interior edges have exactly two different adjacent triangles.

Remark 3.1. *Manifold-like triangulations are typical for many applications of mesh refinement, such as the numerical solution of partial differential equations. In the case* $d = 2$ *this includes triangulations of domains, and in the case* $d = 3$ *this includes triangulated surfaces embedded in Euclidean* 3*-space; see Figure 4. We remark that a manifold-like triangulation may still have vertices around which the triangulation does not look like a manifold.*

We call an edge $e \in \mathcal{E}$ *compatibly bisectable* if it is the refinement edge of all triangles adjacent to e, that is, for all $T \in \mathcal{T}$ with $e \subset T$ we have $\mathcal{R}(T) = e$. We call $T \in \mathcal{T}$ *compatibly bisectable* if its refinement edge $\mathcal{R}(T)$ is compatibly bisectable.

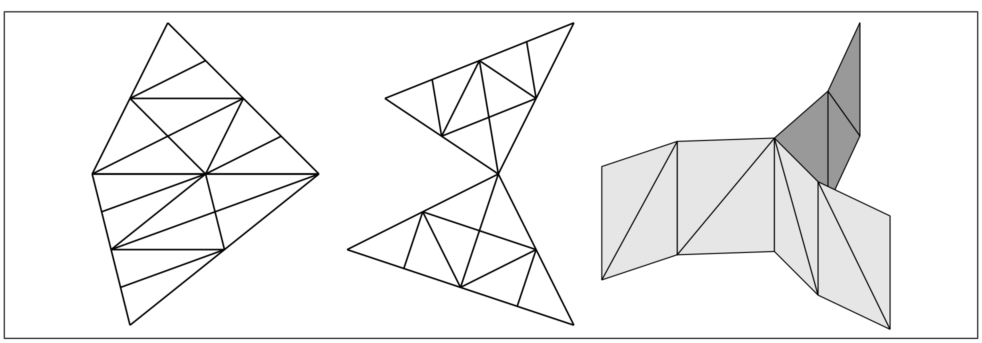


Figure 4. Left and center: manifold-like triangulation, the center having a "singular" vertex. Right: non-manifold-like triangulation

Compatible bisection is the operation of bisecting all triangles sharing a compatibly bisectable edge. Compatible bisections are the fundamental building block of our algorithms.

Note that if e is a border edge, then e is compatible bisectable if and only if it is refinement edge of the single triangle $T \in \mathcal{T}$ that contains e.

If the triangulation is manifold-like, then every interior edge e is an edge of exactly two distinct triangles $T, T' \in \mathcal{T}$, and e is compatibly bisectable if and only if it is the refinement edge of those two triangles T and T'.

Let $\mathcal{T}$ be a triangulation. Let $e \in \mathcal{E}$ be an edge that is compatibly bisectable and let $T_1, T_2, \dots, T_N$ be the list of all triangles adjacent to e. By the definition of compatible bisection, we have $\mathcal{R}(T_i) = e$ for all $1 \leq i \leq N$.

We explicitly describe the modifications resulting from compatible bisection. Recall that when $v_0, v_1 \in \mathbb{R}^n$ are distinct points, then we write $e = [v_0, v_1]$ for the edge between them, and when $v_0, v_1, v_2 \in \mathbb{R}^n$ are affinely independent points, then we write $e = [v_0, v_1, v_2]$ for the triangle spanned by them.

The compatible bisection of e generates a new triangulation $\mathcal{T}'$ by bisecting the triangles $T_1, T_2, \dots, T_N$ from their respective refinement vertices to the midpoint of their common refinement edge. These bisections are newest vertex bisections. More precisely, this works as follows. The edge $e = [v^-, v^+]$ contains two distinct vertices $v^-, v^+ \in \mathcal{V}$. Each triangle $T_i = [v^-, v^+, v_i]$ adjacent to e contains the two vertices v^- and v^+ and a third distinct vertex v_i. Its other two edges are $e_i^- = [v^-, v_i]$ and $e_i^+ = [v^+, v_i]$. The middle point of e is

$$v^0 := \frac{1}{2} v^- + \frac{1}{2} v^+,$$

and we define new edges e^-, e^+, and e_i and new triangles T_i^- and T_i^+ through

$$e^- := [v^-, v^0], \quad e^+ := [v^+, v^0], \quad e_i := [v_i, v^0],$$
$$T_i^- := [v^-, v_i, v^0], \quad T_i^+ := [v^+, v_i, v^0].$$

We then set

$$\mathcal{T}' := (\mathcal{T} \setminus \{T_1, \dots, T_N\}) \cup \{T_1^-, T_1^+, \dots, T_N^-, T_N^+\}.$$

It is clear that $\mathcal{T}'$ is a triangulation. Its set of edges and vertices are

$$\mathcal{V}(\mathcal{T}') = \mathcal{V} \cup \{v^0\}, \quad \mathcal{E}(\mathcal{T}') = (\mathcal{E} \setminus \{e\}) \cup \{e^-, e^+, e^1, \dots, e^N\},$$

respectively. The new assignment of refinement edges $\mathcal{R}' : \mathcal{T}' \to \mathcal{E}(\mathcal{T}')$ is

$$\mathcal{R}'(T) := \begin{cases} \mathcal{R}(T) & \text{if } T \in \mathcal{T} \setminus \{T_1, \dots, T_N\}, \\ e_i^- & \text{if } T = T_i^- \text{ for some } 1 \leq i \leq N, \\ e_i^+ & \text{if } T = T_i^+ \text{ for some } 1 \leq i \leq N. \end{cases}$$

Lemma 3.2. *Let $\mathcal{T}$ be a triangulation, let $e \in \mathcal{E}$ be compatibly bisectable, and let $\mathcal{T}'$ be the triangulation after compatible bisection of e. Then $\mathcal{T}$ is manifold-like if and only if $\mathcal{T}'$ is manifold-like.*

Proof. Compatible bisection along the bisected edge e introduces two new edges, which replace e, and one further edge for each triangle adjacent to e. Each of the two edges replacing e has the same number of adjacent triangles in $\mathcal{T}'$ as e has in $\mathcal{T}$. Each of the new edges dividing a triangle has exactly two adjacent triangles in $\mathcal{T}'$. For each triangle of $\mathcal{T}$ adjacent to e, the two edges that are not bisected are still in $\mathcal{E}(\mathcal{T}')$, with the same number of adjacent triangles. We conclude that $\mathcal{T}$ is manifold-like if and only if $\mathcal{T}'$ is manifold-like. □

Compatible bisections preserve consistency and are geometrically stable since they are based on newest vertex bisection. However, the user may request the bisection of triangles that are not compatibly bisectable. In the next section we explain how to resolve such requests using only compatible bisections.

4. Dependency Graphs

The main result of this section is that any triangle of a triangulation can be refined via a specific sequence of compatible bisections. We utilize graph-theoretical concepts for this endeavor.

For every triangulation $\mathcal{T}$ we define the *triangle dependency graph* $G_T(\mathcal{T})$ as the simple directed graph $G_T(\mathcal{T}) = (\mathcal{N}_T(\mathcal{T}), \mathcal{A}_T(\mathcal{T}))$ whose the set of *nodes* $\mathcal{N}_T(\mathcal{T}) = \mathcal{T}$ is the set of triangles and whose the set of *arrows* is

$$\mathcal{A}_T(\mathcal{T}) = \left\{ (T, T') \in \mathcal{T} \times \mathcal{T} \;\middle|\; T \neq T',\ \mathcal{R}(T) \subset T' \right\}.$$

In this graph, we have an arrow from a triangle T to another triangle T' precisely if the refinement edge of T is an edge of T'.

For every triangulation $\mathcal{T}$ we define the *edge dependency graph* $G_E(\mathcal{T})$ as the simple directed graph $G_E(\mathcal{T}) = (\mathcal{N}_E(\mathcal{T}), \mathcal{A}_E(\mathcal{T}))$ whose set of nodes $\mathcal{N}_E(\mathcal{T}) = \mathcal{R}(\mathcal{T})$ is the set of refinement edges and whose the set of arrows is

$$\mathcal{A}_E(\mathcal{T}) = \left\{ (E, E') \in \mathcal{R}(\mathcal{T}) \times \mathcal{R}(\mathcal{T}) \;\middle|\; E \neq E',\ \exists T \in \mathcal{T} : E \subset T, \mathcal{R}(T) = E' \right\}.$$

In other words, the nodes in the edge dependency graph are the edges $\mathcal{E}$, and we have an arrow from an edge E to another edge E' if E is the refinement edge of some triangle but is also an edge of another triangle T whose refinement edge, however, is a different edge E'.

The compatible bisectability of an edge can be expressed in terms of these graphs. If $e \in \mathcal{R}(\mathcal{T})$ is the refinement edge of some triangle, then e is compatibly bisectable if and only if the triangles in $\mathcal{T}$ which are adjacent to e constitute a clique in the triangle dependency graph.

The description in the edge dependency graph is even simpler. If $e \in \mathcal{R}(\mathcal{T})$ is the refinement edge of some triangle, then e is compatibly bisectable if and only if there are no arrows going out from e in the edge dependency graph. This just means that compatible bisectable edges are precisely the sinks in the directed graph $G_E(\mathcal{T})$.

Remark 4.1. *In a manifold-like triangulation, every edge is shared by at most two triangles. Hence, there is at most one arrow going out from every node in the triangle dependency graph and there is at most one arrow going out from every node in the edge dependency graph.*

Remark 4.2. *We remark that an arrow pointing from a triangle T_1 to another triangle T_2 in the triangle dependency graph corresponds to the situation in newest vertex bisection that to bisect T_1 requires T_2 to bisected, either simultaneously or in* *a preprocessing step. Similarly for the edge dependency graph: an arrow pointing from an edge E_1 to another edge E_2 corresponds to the situation in newest vertex bisection that to bisect along an edge E_1, it is required to bisect along another E_2 as preprocessing.*

We call a triangulation $\mathcal{T}$ *acyclic* if $G_E(\mathcal{T})$ contains no cycles. This is the case if and only if all cycles in $G_T(\mathcal{T})$ are part of a clique. The intuition is that the edge dependency graph is acyclic precisely if there are no cyclic dependencies between the refinement edges.

We state the following two lemmas for completeness, characterizing compatibly bisectable edges and acyclic assignments of refinement edges in terms of graph theoretical properties.

Lemma 4.3. *An edge $e \in \mathcal{E}(\mathcal{T})$ is compatibly bisectable if and only if all triangles adjacent to e form a complete subgraph in $G_T(\mathcal{T})$.*

Proof. By definition, e is compatibly bisectable if and only if it is the refinement edge of all triangles adjacent to e, which precisely means that every triangle adjacent to e depends on every other triangle adjacent to e. This is the case if and only if all triangles adjacent to e form a complete subgraph in $G_T(\mathcal{T})$. □

Lemma 4.4. *The graph $G_E(\mathcal{T})$ is acyclic if and only if all cycles in $G_T(\mathcal{T})$ are part of a complete subgraph. In particular, if $\mathcal{T}$ is manifold-like, then $G_E(\mathcal{T})$ is acyclic if and only if all cycles in $G_T(\mathcal{T})$ have exactly two members.*

Proof. Suppose that $G_E(\mathcal{T})$ contains a cycle $e_0, e_1, \dots, e_m \in \mathcal{R}(\mathcal{T})$ with $e_0 = e_m$. Then there exists a sequence $T_0, \dots, T_m \in \mathcal{T}$ with $\mathcal{R}(T_i) = e_i$ and $T_0 = T_m$ and T_i depends on T_{i+1} for $0 \leq i < m$. If those triangles were a complete subgraph in $G_T(\mathcal{T})$, then those triangles would share their refinement edge with each other, which would imply $e_0 = \dots = e_m$, contradicting our assumption that these edges are a cycle in $G_E(\mathcal{T})$. So $G_T(\mathcal{T})$ must contain a cycle that is not part of a complete subgraph.

Conversely, suppose that $G_E(\mathcal{T})$ is acyclic but that $G_T(\mathcal{T})$ contains a cycle $T_0, \dots, T_m \in \mathcal{T}$ with $T_0 = T_m$ that is not a complete subgraph of $G_T(\mathcal{T})$. Then there are two triangles T_i and T_j that do not depend on each other, so $\mathcal{R}(T_i) \neq \mathcal{R}(T_j)$. But then it follows that there exists a cycle in $G_E(\mathcal{T})$ that contains $\mathcal{R}(T_i)$ and $\mathcal{R}(T_j)$, contradicting $G_E(\mathcal{T})$ being acyclic. Hence every cycle in $G_T(\mathcal{T})$ must be part of a complete subgraph.

The specialization in the case of manifold-like $\mathcal{T}$ is obvious. □

Remark 4.5. *If the triangulation is manifold-like, then $G_E(\mathcal{T})$ has at most one outgoing arrow per node. Hence for a manifold-like triangulation, $G_E(\mathcal{T})$ is acyclic if and only if it is a* branching *in the terminology of graph theory.*

If an assignment of refinement edges is acyclic, then after any compatible bisection the resulting assignment of refinement edges will be acyclic again.

Lemma 4.6. *Let $\mathcal{T}$ be a triangulation. Let $e \in \mathcal{E}$ be compatibly bisectable, and let $\mathcal{T}'$ be the triangulation after compatible bisection of e. If $G_E(\mathcal{T})$ is acyclic, then $G_E(\mathcal{T}')$ is acyclic.*

Proof. Suppose that $G_E(\mathcal{T})$ but that $G_E(\mathcal{T}')$ contains a cycle. Then the edges of that cycle are already in $\mathcal{E}$ since none the new edges introduced by compatible bisection are refinement edges. Furthermore, any triangle of $\mathcal{T}'$ that contains two of the edges in that cycle are already members of $\mathcal{T}$. Hence $G_E(\mathcal{T})$ has a cycle. This contradicts $G_E(\mathcal{T})$ being acyclic, so $G_E(\mathcal{T}')$ must be acyclic too. □

Suppose that $G_E(\mathcal{T})$ is acyclic. If $e, e' \in \mathcal{R}(\mathcal{T})$ are refinement edges such that from e we can reach e' in the edge dependency graph, then we say that e *depends* on e' and that e' is *necessary* for e. Furthermore, if $e, e' \in \mathcal{R}(\mathcal{T})$ are refinement edges such that there is an arrow from e to e' in the edge dependency graph, then we say that e *depends immediately* on e' and that e' is *immediately necessary* for e. In particular, each edge e depends on itself and is necessary for itself. A finite sequence $e_1, \dots, e_N \in \mathcal{R}(\mathcal{T})$ of refinement edges is called a *dependency chain* if for all $1 \leq i \leq N$ the subsequence $e_{i+1}, \dots, e_N$ contains all the refinement edges on which e_i depends. Note that the last edge e_N is compatibly bisectable by definition of dependency chain.

Lemma 4.7. *Let $\mathcal{T}$ be an acyclic triangulation and let $e_1, \dots, e_N \in \mathcal{R}(\mathcal{T})$ be a dependency chain. Then e_N is compatibly bisectable, and $e_1, \dots, e_{N-1} \in \mathcal{R}(\mathcal{T}')$ is a dependency chain in the triangulation $\mathcal{T}'$ that is obtained after compatible bisection along e_N.*

Proof. This follows from definitions. □

We conclude that whenever we want to have a set of edges refined via a sequence of compatible bisections, we should find the dependency chain that contains those marked edges. The next lemma asserts that this is always feasible.

Theorem 4.8. *Let $\mathcal{T}$ be a acyclic triangulation and let $\mathcal{M} \subseteq \mathcal{R}(\mathcal{T})$. Then there exists a unique minimal dependency chain $e_1, \dots, e_N \in \mathcal{R}(\mathcal{T})$ that contains all edges in $\mathcal{M}$.*

Proof. Consider the minimal subgraph of $G_E(\mathcal{T})$ that contains the marked edges $\mathcal{M}$ and all the edges reachable from $\mathcal{M}$ within $G_E(\mathcal{T})$. This subgraph is acyclic. Topological sorting of its nodes then produces the required dependency chain. □

In implementations we only perform the minimal refinements for any set of marked edges. Suppose we have an acyclic triangulation $\mathcal{T}$. Whenever $\mathcal{M} \subseteq \mathcal{E}(\mathcal{T})$ is a set of marked edges, we write

$$\mathcal{T}' := \textsc{Refine}\,(\mathcal{T}, \mathcal{M})$$

for the triangulation $\mathcal{T}'$ that is obtained from the processing the minimal dependency chain associated to $\mathcal{M}$.

Lemma 4.9. *Let $\mathcal{T}$ be a triangulation and let $\mathcal{U} \subseteq \mathcal{T}$ be a subtriangulation. If $\mathcal{T}'$ is a triangulation constructed from repeated recursive refinement starting with $\mathcal{T}$, then the descendants of triangles in $\mathcal{U}$ can be constructed from repeated recursive refinement starting with $\mathcal{U}$.*

Proof. This is obvious. □

5. Algorithms for Global Mesh Refinement

Suppose that we have an acyclic triangulation $\mathcal{T}$. As we have shown in the preceding section, for any set $\mathcal{M} \subseteq \mathcal{E}(\mathcal{T})$ of marked edges we can find a unique minimal sequence of compatible bisections that will have that set of edges refined.

The compatible bisections preserve the consistency of the mesh, and repeated application of this mark-refine procedure is geometrically stable. In this section, we outline a few possible implementations of the refinement step.

The considerations so far imply that any set of marked edges is eventually refined if we keep bisecting compatibly bisectable edges necessary for those marked edges; see Algorithm 1 for an abstract pseudocode. By construction, the algorithm terminates, and we will have bisected only those edges necessary for the refinement of the marked edges.

Algorithm 1 Global Bisection Algorithm

procedure Refine($\mathcal{T}$, $\mathcal{M} \subseteq \mathcal{R}(\mathcal{T})$)
  while $\mathcal{M} \neq \emptyset$ **do**
    Pick $e_M \in \mathcal{M}$.
    Let $e \in \mathcal{E}$ be compatible bisectable and necessary for e_M.
    Bisect along e and let $\mathcal{M} := \mathcal{M} \setminus \{e\}$.

An alternative form of Refine constructs the entire dependency chain of the marked edges prior to starting the sequence of refinements; see Algorithm 2. We can construct a dependency chain $e_1, \dots, e_N$ that contains any arbitrary set $\mathcal{M} \subseteq \mathcal{R}(\mathcal{T})$ of marked edges using some subprocedure, such as breadth-first search in the edge dependency graph starting from the edges in $\mathcal{M}$. Then all marked edges will have been refined after working through the dependency chain.

Algorithm 2 Global Bisection Algorithm (alternative version)

procedure Refine($\mathcal{T}$, $\mathcal{M} \subseteq \mathcal{R}(\mathcal{T})$)
  Construct minimal dependency chain $e_1, \dots, e_N$ for $\mathcal{M}$.
  for k from N down to 1 **do**
    Bisect along e_k.

Finally, we give an recursive form of Refine; see Algorithm 3. Conceptually, the algorithm travels from any marked edge to a sink of the edge dependency graph and then returns to the marked edges, performing compatible bisections along the way. It requires a subroutine to determine whether a marked edge has already been refined in prior operations.

Algorithm 3 Global Bisection Algorithm (recursive form)

procedure Refine($\mathcal{T}$, $\mathcal{M} = \{e_1, \dots, e_M\} \subseteq \mathcal{R}(\mathcal{T})$)
  for m from 1 to M **do**
    if e_m has not been bisected yet **then**
      RecursiveRefinement($\mathcal{T}$, e_m)

procedure RecursiveRefinement($\mathcal{T}$, e)
  for $e' \in \mathcal{R}(\mathcal{T}) \setminus \{e\}$ necessary for e and sharing a triangle with e **do**
    RecursiveRefinement($\mathcal{T}$, e')
  Perform compatible bisection of e

In most applications it is the triangles that are marked for refinement and the number of triangles created is the quantity that is of interest. Therefore our amortized complexity will be given in terms of the number of triangles marked and refined. We “overload” RecursiveRefinement as follows:

Algorithm 4 Global Bisection Algorithm (recursive form, triangles marked)

```
1: procedure Refine( 𝒯, 𝒰 ⊆ 𝒯 )
2:     Let ℰ = {e ∈ ℰ(𝒯) | ∃T ∈ 𝒯 : ℛ(T) = e}
3:     Refine(𝒯, ℰ)
```

6. Amortized Complexity

Recursive newest vertex bisection not only refines the marked triangles but also performs compatible bisections along additional edges. In this section we derive an amortized complexity estimate that bounds the number of additional bisections: we show that the total number of triangles bisected is bounded by the number of triangles marked for refinement.

We consider a sequence of triangulations that is recursively defined by the application of Refine with a sequence of marked edges. Formally, starting with the initial triangulation $\mathcal{T}_0$ we define a sequence of triangulations $\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2, \dots$ by

$$\mathcal{T}_{i+1} := \textsc{Refine}\left(\mathcal{T}_i, \mathcal{M}_i\right), \quad i \in \mathbb{N}_0,$$

where $\mathcal{M}_i \subseteq \mathcal{E}(\mathcal{T}_i)$ is a set of edges marked in the i-th triangulation.

6.1. Levels and Combinatorial Results.

We first develop several combinatorial results that relate to the notion of level of triangles. In this subsection, we fix a triangulation $\mathcal{T}$ that has evolved from repeatedly invoking recursive refinement, starting from the initial triangulation $\mathcal{T}_0$.

The *level* of a triangle is defined recursively as follows. All triangles $T \in \mathcal{T}_0$ in the initial triangulation are said to have level $\ell(T) = 0$. If a triangle T' is a child of the triangle T, then the level of T' is $\ell(T') := \ell(T) + 1$. In other words, the children of a triangle have one level more than the parent triangle.

We say that a triangulation is *ideally matched* if all triangles in that triangulation are compatibly bisectable.

Remark 6.1. *A triangulation of a closed compact surface without boundary is ideally matched if the assignment of refinement edges describes a perfect matching in the adjacency graph of the triangulation. Triangles with a boundary edge may have that edge selected as refinement edge.*

Lemma 6.2. *Let $\mathcal{T}_0$ be ideally matched and let $T, S \in \mathcal{T}$ be adjacent. Assume that T immediately depends on S. If S does not immediately depend on T, then $\ell(T) = \ell(S) + 1$. Otherwise $\ell(T) = \ell(S)$.*

Proof. It suffices to show that this property is invariant under compatible bisections. Let $e \in \mathcal{R}(\mathcal{T})$ be the edge along which we bisect. Let $T_1, \dots, T_N$ be the triangles adjacent to e and let $T_1^+, T_1^-, \dots, T_N^+, T_N^-$ be the respective child triangles after compatible bisection. Then $\ell(T_1) = \cdots = \ell(T_N)$ and for all $1 \leq i \leq N$ we have $\ell(T_i^-) = \ell(T_i^+) = \ell(T_i) + 1$. By construction, adjacent children of the original triangles have the same level and do not depend on each other.

Let $S \in \mathcal{T}$ be adjacent to one of the triangles T_i but not share the edge e. Then S will be adjacent to a child of T_i, say, to T_i^+. If S immediately depends on T_i, then $\ell(S) = \ell(T_i) + 1$, and S will be in mutual dependence with T_i^+ with whom it shares the same level. If S does not immediately depend on T_i, then $\ell(S) = \ell(T_i)$ and one of the children will have a non-reciprocal dependence on S with $\ell(S) + 1 = \ell(T_i^+)$. This shows the desired result. □

Lemma 6.3. *Let $\mathcal{T}_0$ be ideally matched. Let $S, T \in \mathcal{T}$ such that T immediately depends on S. Then $\ell(T) \geq \ell(S)$ and the descendants of S that appear after compatible bisection of T have level at most $\ell(T) + 1$.*

Proof. The inequality $\ell(T) \geq \ell(S)$ follows from Lemma 6.2. Let S' be a descendant of S that appears during recursive refinement. If S has been compatibly bisectable in $\mathcal{T}$, then S' is a child of S and thus $\ell(S') = \ell(S) + 1 \leq \ell(T) + 1$. Otherwise S' is a child or grandchild of S. Then $\ell(S') \leq \ell(S) + 2 \leq \ell(T) + 1$. □

We now prove several auxiliary results with the goal of proving an analogue of Lemma 6.3 for when the initial triangulation $\mathcal{T}_0$ is not ideally matched. The following lemmata can be found in [24] and are included to keep the presentation self-contained.

Lemma 6.4. *Let $T, S \in \mathcal{T}$ be adjacent descendants of the same triangle in $\mathcal{T}_0$. Assume that T immediately depends on S. If S does not immediately depend on T, then $\ell(T) = \ell(S) + 1$. Otherwise $\ell(T) = \ell(S)$.*

Proof. Let $T_0 \in \mathcal{T}_0$ be the ancestor of S and T. Then $\mathcal{U}_0 = \{T_0\}$ is ideally matched. If S and T are descendants of T_0 via recursive refinement over $\mathcal{T}_0$, then they are also descendants of T_0 via recursive refinement over $\mathcal{U}_0$. The claim now follows from Lemma 6.2. □

Lemma 6.5. *Let $T_0, S_0 \in \mathcal{T}_0$ be adjacent such that the shared edge of T_0 and S_0 is either the refinement edge of both or none of them. Let $T, S \in \mathcal{T}$ be adjacent descendants of T_0 and S_0, respectively. Assume that T immediately depends on S. If S does not immediately depend on T, then $\ell(T) = \ell(S) + 1$. Otherwise $\ell(T) = \ell(S)$.*

Proof. The first compatible bisection performed on S_0 or T_0 must be the compatible bisection of S, producing a child S' that is compatibly bisectable with T. Consider the triangulation $\mathcal{U}_0 = \{S', T\}$. Descendants of S adjacent to T are descendants of S'. We introduce a new level function ℓ' on $\mathcal{U}_0$ such that $\ell'(S') = \ell'(T) = 0$. Since $\mathcal{U}_0$ is ideally matched, we use Lemma 6.2 together with $\ell'(S) + 1 = \ell(S)$. □

Lemma 6.6. *Let $T_0, S_0 \in \mathcal{T}_0$ be adjacent such that T_0 immediately depends on S_0 but not vice versa. Let $T, S \in \mathcal{T}$ be adjacent descendants of T_0 and S_0, respectively. If T immediately depends on S but not vice versa, then $\ell(T) = \ell(S)$. If instead S immediately depends on T but not vice versa, then $\ell(S) = \ell(T) + 2$. Otherwise, $\ell(T) = \ell(S) - 1$.*

Proof. Let T_0 and S_0 be the ancestors of T and S, respectively, in the initial mesh $\mathcal{T}_0$. Then S and T are adjacent. We see that S must be a descendant of a child of S_0, say, S_0^+. Then $T, S \in \mathcal{T}$ appear after newest vertex bisection applied to the mesh $\mathcal{U} := \{T_0, S_0^+\}$ which is ideally matched. Let us introduce a new notion of level g on the mesh $\mathcal{U}$. Then $g(T) = \ell(T)$ and $g(S) + 1 = \ell(S)$.

We have $g(T) = g(S)$ if and only if the edge shared by S and T is either the refinement edge of both of them or neither of them, and in that case we $\ell(T) = \ell(S) - 1$.

If S immediately depends on T but not vice versa, then $g(S) = g(T) + 1$, and hence $\ell(S) - 1 = \ell(T) + 1$, that is, $\ell(S) = \ell(T) + 2$. If instead T immediately depends on S but not vice versa, then $g(T) = g(S) + 1$, that is, $\ell(T) = \ell(S)$. □

We now have an analogue of Lemma 6.3 for general initial triangulations $\mathcal{T}_0$.

Lemma 6.7. *Let $S, T \in \mathcal{T}$ such that T depends on S. Then $\ell(T) + 1 \geq \ell(S)$ and the descendants of S that appear after recursive refinement have level at most $\ell(T) + 2$.*

Proof. A careful inspection of Lemmas 6.4, 6.5, and 6.6 immediately shows the inequality $\ell(T) + 1 \geq \ell(S)$.

Let S' be a descendant of S that appears during recursive refinement of T. If S has been compatibly bisectable in $\mathcal{T}$, then S' is a child of S. In that case, $\ell(S') = \ell(S) + 1 \leq \ell(T) + 2$. If S has not been compatibly bisectable in $\mathcal{T}$, then S' is a child or grandchild of S. Another inspection of Lemmas 6.4, 6.5, and 6.6 shows that $\ell(S) \leq \ell(T)$ and thus $\ell(S') \leq \ell(T) + 2$. □

It is evident that the level of triangles does not increase along a refinement chain (with the possible exception of the very end). We would like to quantify to which extent the level stays constant throughout the chain. The following lemma quantitatively strengthens a result by Karkulik, Pavlicek and Preatorius [24, Proposition 6.vii].

Lemma 6.8. *Let $T_1, \dots, T_k \in \mathcal{T}$ be a sequence of triangles such that for all $2 \leq i \leq k$ we have $\ell(T_i) = \ell(T_{i-1})$ and T_{i-1} immediately depending on T_i but not vice versa.*

Then k is at most the length of the longest dependency chain in the initial triangulation. Furthermore, if $\ell(T_1) > 0$ and $k \geq 3$, then all triangles T_i share a common vertex $v \in \mathcal{V}(\mathcal{T}_0)$ and k is at most the valency of v in $\mathcal{T}_0$.

Proof. If $\ell(T_1) = 0$, then obviously k is at most the maximal length of any refinement sequence in the initial triangulation. So it remains to consider the case $\ell(T_1) > 0$.

Let $S_i \in \mathcal{T}_0$ be the ancestor of T_i in the original triangulation. Suppose that $S_{i-1} = S_i$ for some $2 \leq i \leq k$. If $\ell(T_{i-1}) = \ell(T_i)$. then Theorem 4.9 and Lemma 6.4 imply a contradiction. Thus consecutive triangles in the sequence must be descendants of different ancestors in $\mathcal{T}_0$. By Lemma 6.5 and Lemma 6.4, we see that S_{i-1} immediately depends on S_i but not vice versa. Hence k must be at most the maximal length of any refinement sequence in the initial triangulation.

Now assume $k \geq 3$. We see that the triangles $T_2, \dots, T_{k-1}$ each have two different edges that are contained in edges of $\mathcal{E}(\mathcal{T}_0)$ and thus each has a vertex contained in $\mathcal{V}(\mathcal{T}_0)$. Suppose that one of the triangles has two vertices in $\mathcal{V}(\mathcal{T}_0)$. Then $\ell(T_i) = 1$ because $\ell(T_i) > 0$ by assumption, and the aforementioned edge $e \in \mathcal{E}(\mathcal{T}_0)$ is the refinement edge of T_i. But e is not the refinement edge of T_{i+1} by assumption. We see that the parents of T_i and T_{i+1} in $\mathcal{T}_0$ must not have been mutually dependent.

Then $\ell(T_i) > \ell(T_{i+1})$ by Lemma 6.5 and a contradiction follows. We conclude that each T_i with $2 \leq i \leq k-1$ has at most one vertex from $\mathcal{V}(\mathcal{T}_0)$. It is easily seen that this vertex $v \in \mathcal{V}(\mathcal{T}_0)$ is common to every T_i, $1 \leq i \leq k$. This yields the desired bound. □

6.2. **Triangulation Parameters and an Auxiliary Lemma.** The remaining auxiliary lemmas of this section and also our main result involve several parameters that depend on combinatorial and geometric properties of the triangulation.

We introduce a constant $\zeta \in \{0,1\}$ that we set to $\zeta = 0$ if $\mathcal{T}_0$ is ideally matched and that we set to $\zeta = 1$ otherwise. This constant is essentially a 'binary flag' and allows us to improve the bounds in the main result.

We fix constants $0 < D_1 \leq D_2$ such that for all triangles $T \in \mathcal{T}$ we have

$$D_1 2^{-\ell(T)/2} \leq \operatorname{meas}(T)^{\frac{1}{2}} \leq \operatorname{diam}(T) \leq D_2 2^{-\ell(T)/2}. \tag{5}$$

For example, we may choose

$$D_1 = \min_{T_0 \in \mathcal{T}_0} \operatorname{meas}(T_0)^{\frac{1}{2}}, \quad D_2 = 2 \max_{T_0 \in \mathcal{T}_0} \operatorname{diam}(T_0),$$

in accordance with Lemma 2.2.

We will discuss a few definitions and results that refer to the distance between two triangles. The distance $\operatorname{dist}(S,T)$ between triangles S and T is the infimum of the lengths of all rectifiable paths from S to T that are contained in $\mathcal{T}_0$.

We assume that for any $\delta \geq 0$ we have a constant $C_{\sharp,\delta} > 0$ such for any $T' \in \mathcal{T}$ for all $1 \leq k \leq \ell(T') + \zeta + 1$ we have

$$|\{ S \in \mathcal{T} \mid \operatorname{dist}(S,T') \leq \delta 2^{-k/2}, \ell(S) = k \}| \leq C_{\sharp,\delta}. \tag{6}$$

An illustration for this technical definition is provided below. The existence of the constant $C_{\sharp,\delta} > 0$ is clear from the following intuition: if $\operatorname{dist}(S,T') \leq \delta 2^{-k/2}$ and $\ell(S) = k$, then S is contained in a ball around T' of radius comparable to $\delta^2 2^{-k}$. While that ball has area comparable to $\delta^2 2^{-k}$, the triangle S as area comparable to 2^{-k}. So the set in (6) must have uniformly bounded cardinality.

To make this intuitive notion rigorous and bound the constant $C_{\sharp,\delta}$, we prove the following two auxiliary results.

Lemma 6.9. *For $R > 0$ small enough there exists a constant $C_{\pi,R} > 0$ such that any ball B of radius at most R within $\mathcal{T}_0$ satisfies the inequality*

$$\operatorname{meas}(B) \leq C_{\pi,R} R^2.$$

For all $R > 0$ we have $C_{\pi,R} < \infty$ for each triangulation.

Proof. Let $\epsilon > 0$ be the shortest edge length of $\mathcal{T}$. Let $C > 0$ such that every triangle $T \in \mathcal{T}_0$ is sharing a vertex with at most C triangles from $\mathcal{T}_0$. Let x be a point in the triangulation and let $r \in [0,\epsilon)$. Then the set $B_r(x)$ contains points of at most C different triangles and the intersection of $B_r(x)$ and each of those triangles has measure at most πr^2. Hence $\operatorname{meas}(B_r(x)) \leq C\pi r^2$. Hence $C_{\pi,R} = \max\{C\pi r^2, \operatorname{meas}(\mathcal{T})/\epsilon^2\}$ is the upper bound. □

Remark 6.10. *For example, we have $C_{\pi,\delta} = \pi$ if the triangulation is embedded in $\mathbb{R}^2$. Generally $C_{\pi,\delta}$ might be larger though.*

Lemma 6.11. *For all $\delta \geq 0$ we have $C_{\sharp,\delta} < \infty$.*

Proof. Let $T \in \mathcal{T}$. Suppose that $S \in \mathcal{T}$ satisfies

$$\ell(S) \leq \ell(T) + 1 + \zeta, \quad \operatorname{dist}(S,T) \leq \delta 2^{-\ell(S)/2}.$$

Recall that

$$\operatorname{meas}(S) \geq D_1^2 2^{-\ell(S)}, \quad \operatorname{diam}(S) \leq D_2 2^{-\ell(S)/2}, \quad \operatorname{diam}(T) \leq D_2 2^{-\ell(T)/2}.$$

Evidently, S is contained within a radius of $\operatorname{diam}(T) + \delta 2^{-\ell(S)/2} + \operatorname{diam}(S)$ around any point of T. We estimate

$$\begin{aligned} &\operatorname{diam}(T) + \delta 2^{-\ell(S)/2} + \operatorname{diam}(S) \\ &\leq D_2 2^{-\ell(S)/2+1/2+\zeta/2} + \delta 2^{-\ell(S)/2} + D_2 2^{-\ell(S)/2} \\ &\leq \left(2_{1/2+\zeta/2} D_2 + \delta + D_2\right) 2^{-\ell(S)/2}. \end{aligned}$$

Hence that ball's volume is at most $C_{\pi,\rho}\rho^2 2^{-\ell(S)}$ with $\rho = 2^{1/2+\zeta/2} D_2 + \delta + D_2$. We conclude that $C_{\sharp,\delta} \leq C_{\pi,\delta}\rho^2/D_1^2$. □

Throughout the derivation of the main result, we use a few purely numerical constants. Here and below, we abbreviate

$$C_\Sigma := \sum_{i=0}^{\infty} 2^{-i/2} = \frac{1}{1 - 1/\sqrt{2}}.$$

We also assume to have two sequences of positive real numbers $(a_k)_{k\geq -1-\zeta}$ and $(b_k)_{k\geq 0}$ satisfying $b_0 \geq 1$ and

$$A := \sum_{k=-1-\zeta}^{\infty} a_k < \infty, \quad B := \sum_{k=0}^{\infty} 2^{-k/2} b_k < \infty, \quad \gamma := \inf_{k\in\mathbb{N}_0} a_k b_k > 0. \tag{7}$$

Remark 6.12. *The sequences can be chosen as* $a_k = (k+2+\zeta)^{-2}$ *and* $b_k = 2^{\frac{k}{4}}$. *Then* $A = \pi^2/6$, $B = (1-2^{-1/4})^{-1}$, *and* $\gamma \geq 3/100$. *This choice of parameters has appeared in the literature with minor variations (see* [6, 36, 31, 24]*) but is by no means the only possible choice.*

The following auxiliary result states that any triangle is only necessary for other triangles when they are not too far away.

Theorem 6.13. *Let* $T, S \in \mathcal{T}$ *such that* T *depends on* S. *Let* S' *be either* S, *a child of* S, *or a grandchild of* S *that is constructed by invoking recursive refinement on* T. *Then*

$$\operatorname{dist}(S',T) \leq C_{\mathrm{D}} 2^{-\ell(S')/2}.$$

Here, $C_{\mathrm{D}} := D_2 C_{\mathrm{L}} C_\Sigma$.

Proof. There exists a sequence of pairwise distinct triangles $T_0, T_1, \ldots, T_k$ such that $T_0 = S$ and $T_k = T$ and such that T_i immediately depends on T_{i-1} for $1 \leq i \leq k$. Let $z_i \in T_i$ be the midpoint of the refinement edge of T_i for $0 \leq i \leq k$. Note that $z_i \in S$ is adjacent to every grandchild of T_i.

For any $1 \leq i \leq k$ the segment from z_i to z_{i-1} is an edge of some grandchild S_{i-1} of T_{i-1}. For any $j \in \{0,1\}$ we thus see

$$\begin{aligned}
\operatorname{dist}(z_j, z_k) &\leq \sum_{i=j+1}^{k} \operatorname{dist}(z_{i-1}, z_i) \\
&\leq \sum_{i=j+1}^{k} \operatorname{diam}(S_{i-1}) \leq \sum_{i=j}^{k-1} \operatorname{diam}(S_i) \leq D_2 \sum_{i=j+1}^{k} 2^{-\ell(S_{i-1})/2}
\end{aligned}$$

Let $j \in \{0,1\}$ be such that $\ell(T_i) \leq \ell(T_{i+1})$ for any $j \leq i < k$. We then have $\ell(S_i) \leq \ell(S_{i+1})$ for any $j \leq i < k$. Note that $j = 1$ is a valid choice since we assume that T_i does not depend on T_{i+1} for $1 \leq i \leq k-1$. For any such choice of j, we have

$$\begin{aligned}
\operatorname{dist}(z_j, z_k) &\leq \sum_{i=j}^{k-1} \operatorname{diam}(S_i) \\
&\leq \sum_{l=\ell(S_j)}^{\infty} \sum_{\substack{j \leq i \leq k-1 \\ \ell(S_i)=l}} \operatorname{diam}(S_i) \leq D_2 \sum_{l=0}^{\infty} \sum_{\substack{j \leq i \leq k-1 \\ \ell(S_i)=l}} 2^{-(\ell(S_j)+l)/2} \\
&\leq D_2 C_{\mathrm{L}} 2^{-\ell(S_j)/2} \sum_{l=0}^{\infty} 2^{-l/2} \leq D_2 C_{\mathrm{L}} C_{\Sigma} 2^{-\ell(S_j)/2}.
\end{aligned}$$

Consider the case that T_0 is not compatibly bisectable with T_1. Then $j = 0$ is a valid choice and all grandchildren of T_0 are adjacent to z_0. Together with $\ell(S') \leq \ell(S_0)$ we get

$$\operatorname{dist}(S', T) \leq D_2 C_{\mathrm{L}} C_{\Sigma} 2^{-\ell(S_0)/2} \leq D_2 C_{\mathrm{L}} C_{\Sigma} 2^{-\ell(S')/2}.$$

Now consider the case that T_0 is compatibly bisectable with T_1. Then $S' = S$ or S' is a child of S. Then all grandchildren of T_0 are adjacent to z_1, and so $\operatorname{dist}(S', T) \leq \operatorname{dist}(z_1, z_k)$. We have $\ell(T_0) = \ell(T_1)$ or $\ell(T_0) = \ell(T_1) + 1$. In either case,

$$\ell(S') \leq \ell(T_0) + 1 \leq \ell(T_1) + 2 \leq \ell(S_1).$$

This shows that

$$\operatorname{dist}(z_1, z_k) \leq D_2 C_{\mathrm{L}} C_{\Sigma} 2^{-\ell(S_1)/2} \leq D_2 C_{\mathrm{L}} C_{\Sigma} 2^{-\ell(S')/2},$$

which is the desired estimate. □

Corollary 6.14. *Let $S, T \in \mathcal{T}$ such that T depends on S. Then* $\operatorname{dist}(T, S) \leq C_{\mathrm{D}}$.

Remark 6.15. *The preceding corollary has an immediate relevance to the implementation of newest vertex bisection in a parallel setting with distributed memory. It restricts the physical range of dependency chains.*

We now have the machinery to prove the amortized complexity estimate of our mesh refinement algorithm.

Theorem 6.16. *There exists $\Lambda > 0$, depending only on $\mathcal{T}_0$, such that for all sequences $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_g$ of sets of triangles marked for refinement we have*

$$|\mathcal{T}_g| - |\mathcal{T}_0| \leq \Lambda \sum_{i=0}^{g-1} |\mathcal{M}_i|. \tag{8}$$

Explicitly, we have

$$\Lambda \leq \frac{C_{\sharp,\delta} A}{\min\{a_{-1}, a_{-1-\zeta}, a_0, \gamma\}}, \quad \delta := (C_{\mathrm{D}} + D_2)B.$$

Proof. Let us write $\mathcal{M} = \mathcal{M}_0 \cup \mathcal{M}_1 \cup \dots \cup \mathcal{M}_g$. We construct a function

$$\lambda : (\mathcal{T}_g \setminus \mathcal{T}_0) \times \mathcal{M} \to \mathbb{R} \tag{9}$$

such that

$$\forall T' \in \mathcal{M} : \sum_{T \in \mathcal{T}_g \setminus \mathcal{T}_0} \lambda(T, T') \leq C^{\$}, \tag{10}$$

$$\forall T \in \mathcal{T}_g \setminus \mathcal{T}_0 : \sum_{T' \in \mathcal{M}} \lambda(T, T') \geq C_{\$}, \tag{11}$$

It will then follow that

$$C_{\$} \left(|\mathcal{T}_g| - |\mathcal{T}_0| \right) \leq \sum_{T \in \mathcal{T}_g \setminus \mathcal{T}_0} \sum_{T' \in \mathcal{M}} \lambda(T, T') \leq \sum_{T' \in \mathcal{M}} C^{\$} \leq |\mathcal{M}| \cdot C^{\$}.$$

Thus $\Lambda := C^{\$}/C_{\$}$ is the constant that we are looking for. Hence it remains to find the function λ satisfying the upper and lower bounds (10) and (11).

We define the quantity

$$\delta := (C_{\mathrm{D}} + D_2)B.$$

The aforementioned function λ is defined by

$$\lambda(T, T') = \begin{cases} a_{\ell(T') - \ell(T)} & \text{if } \operatorname{dist}(T, T') < \delta 2^{-\ell(T)/2} \text{ and } \ell(T) \leq \ell(T') + 1 + \zeta, \\ 0 & \text{otherwise.} \end{cases}$$

In particular, for $T \in \mathcal{T}$ and $T' \in \mathcal{M}$ we have $\lambda(T, T') > 0$ only if T has level at most $\ell(T') + 1 + \zeta$.

We prove the upper bound (10). Fix $T' \in \mathcal{M}$. The definition of λ then gives

$$\sum_{T \in \mathcal{T}_g \setminus \mathcal{T}_0} \lambda(T, T') = \sum_{k=1}^{\ell(T')+1+\zeta} \sum_{\substack{T \in \mathcal{T}_g \setminus \mathcal{T}_0 \\ \ell(T) = k}} \lambda(T, T') = \sum_{k=1}^{\ell(T')+1+\zeta} \sum_{\substack{T \in \mathcal{T}_g \setminus \mathcal{T}_0 \\ \ell(T) = k}} a_{\ell(T') - k}.$$

If $T \in \mathcal{T}_g \setminus \mathcal{T}_0$ with $\ell(T) \leq \ell(T') + 1 + \zeta$ and $\lambda(T, T') > 0$, then $\operatorname{dist}(T, T') < \delta 2^{-\ell(T)/2}$. Evidently, using Theorem 6.11, we have

$$\begin{aligned} \sum_{T \in \mathcal{T}_g \setminus \mathcal{T}_0} \lambda(T, T') &\leq C_{\sharp,\delta} \sum_{k=1}^{\ell(T')+1+\zeta} a_{\ell(T') - k} \\ &\leq C_{\sharp,\delta} \sum_{k=-1-\zeta}^{\ell(T')-1} a_k \leq C_{\sharp,\delta} \sum_{k=-1-\zeta}^{\infty} a_k = C_{\sharp,\delta} A =: C_{\$}, \end{aligned}$$

which is the desired upper bound.

Next we prove the lower bound (11). We fix $T_0 \in \mathcal{T}_g \setminus \mathcal{T}_0$. By assumption, there exists a triangle $T_1 \in \mathcal{M}$ such that T_0 has been created by invoking the global mesh refinement algorithm on T_1. We iterate this construction. Suppose that we have defined $T_0, T_1, \dots, T_i$, then either $T_i \in \mathcal{T}_0$ or there exists a triangle $T_{i+1} \in \mathcal{M}$ such that T_i has been created by invoking the global mesh refinement algorithm on T_{i+1}. In this manner, we construct triangles $T_0, T_1, \dots, T_J$, where finally $T_J \in \mathcal{T}_0$ is in the original triangulation.

We have $\ell(T_J) < \ell(T_0)$ since $T_0 \notin \mathcal{T}_0$ but $T_J \in \mathcal{T}_0$. Hence there exists a minimal index $1 \leq s \leq J$ such that $\ell(T_s) < \ell(T_0)$. We have $\ell(T_0) \leq \ell(T_i)$ for $0 \leq i < s$ by definition. According to Lemma 6.3 and Lemma 6.7, for all $0 \leq i < J$ we have $\ell(T_i) \leq \ell(T_{i+1}) + 1 + \zeta$. Therefore,

$$\ell(T_s) < \ell(T_0) \leq \ell(T_{s-1}) \leq \ell(T_s) + 1 + \zeta$$

We conclude that $\ell(T_s) - \ell(T_0) \in \{-1, -1-\zeta\}$.

By definition, for any $1 \leq j \leq J$ we have $\lambda(T_0, T_j) > 0$ only if $\operatorname{dist}(T_0, T_j) < \delta 2^{-\ell(T_0)/2}$ and $\ell(T_0) \leq \ell(T_j) + 1 + \zeta$. We now bound the distance $\operatorname{dist}(T_0, T_j)$ from above.

For $1 \leq j \leq s$ and $l \in \mathbb{N}_0$ we define

$$m(l, j) := |\{\, 0 \leq i \leq j-1 \mid \ell(T_i) = \ell(T_0) + l \,\}| \,. \tag{12}$$

In other words, $m(l, j)$ is the number of those triangles among the $T_0, \cdots, T_{j-1}$ whose level equals $\ell(T_0) + l$.

By the triangle inequality and an induction argument,

$$\operatorname{dist}(T_0, T_j) \leq \sum_{i=1}^{j} \operatorname{dist}(T_{i-1}, T_i) + \sum_{i=1}^{j-1} \operatorname{diam}(T_i).$$

Using (5) and Lemma 6.13, we get

$$\begin{aligned} \operatorname{dist}(T_0, T_j) &\leq C_{\mathrm{D}} \sum_{i=1}^{j} 2^{-\ell(T_{i-1})/2} + D_2 \sum_{i=1}^{j-1} 2^{-\ell(T_i)/2} \\ &\leq C_{\mathrm{D}} \sum_{i=0}^{j-1} 2^{-\ell(T_i)/2} + D_2 \sum_{i=1}^{j-1} 2^{-\ell(T_i)/2}. \end{aligned}$$

Since $0 \leq j \leq s-1$ we have $\ell(T_i) \geq \ell(T_0)$ for all $0 \leq i \leq j-1$, the definition of $m(i, j)$ gives

$$\begin{aligned} \operatorname{dist}(T_0, T_j) &\leq (C_{\mathrm{D}} + D_2) \sum_{i=0}^{j-1} 2^{-\ell(T_i)/2} \\ &= (C_{\mathrm{D}} + D_2) \sum_{l=0}^{\infty} m(l, j) 2^{-(\ell(T_0)+l)/2} \\ &= (C_{\mathrm{D}} + D_2) 2^{-\ell(T_0)/2} \sum_{l=0}^{\infty} m(l, j) 2^{-l/2}. \end{aligned}$$

Here conduct a case distinction based on the size of $m(l, j)$.

Case 1: Consider the case that $m(l,s) \leq b_l$ for all $l \in \mathbb{N}_0$. We then find

$$\begin{aligned}\operatorname{dist}(T_0, T_s) &\leq (C_{\mathrm{D}} + D_2) 2^{-\ell(T_0)/2} \sum_{l=0}^{\infty} m(l,s) 2^{-l/2} \\ &\leq (C_{\mathrm{D}} + D_2) 2^{-\ell(T_0)/2} \sum_{l=0}^{\infty} b_l 2^{-l/2} \\ &\leq (C_{\mathrm{D}} + D_2) B \cdot 2^{-\ell(T_0)/2} \\ &\leq \delta 2^{-\ell(T_0)/2}.\end{aligned}$$

This bound on $\operatorname{dist}(T_0, T_s)$, the condition $\ell(T_s) - \ell(T_0) \in \{-1, -1-\zeta\}$, and the definition of λ now imply

$$\sum_{T \in \mathcal{M}} \lambda(T_0, T) \geq \lambda(T_0, T_s) = a_{\ell(T_s)-\ell(T_0)} \geq \min\{a_{-1}, a_{-1-\zeta}\} > 0.$$

Case 2: Consider the case that there exists a level difference $\ell \in \mathbb{N}_0$ such that $m(\ell, s) > b_\ell$. For each such level difference ℓ, there exists a minimal index $j(\ell)$ such that $m(\ell, j(\ell)) > b_\ell$. We let ℓ^* be the level difference for which the index $j(\ell^*)$ is minimal, and we write $j^* = j(\ell^*)$. Note that we have $m(\ell^*, s) > 1$ since $b_{\ell^*} > 0$ by assumption on the sequence, and that by definition of $m(\ell^*, j^*)$ we have $j^* \geq 1$. By construction, $m(\ell^*, j^*) > b_{\ell^*}$ and

$$m(\ell, j^* - 1) \leq b_\ell, \quad \ell \geq 0.$$

For any $1 \leq i \leq j^* - 1$ such that $\ell(T_i) = \ell(T_0) + \ell$ we now get

$$\begin{aligned}\operatorname{dist}(T_0, T_i) &\leq (C_{\mathrm{D}} + D_2) 2^{-\ell(T_0)/2} \sum_{l=0}^{\infty} m(l,i) 2^{-l/2} \\ &\leq (C_{\mathrm{D}} + D_2) 2^{-\ell(T_0)/2} \sum_{l=0}^{\infty} m(l, j^*-1) 2^{-l/2} \\ &\leq (C_{\mathrm{D}} + D_2) 2^{-\ell(T_0)/2} \sum_{l=0}^{\infty} b_l 2^{-l/2} \\ &\leq (C_{\mathrm{D}} + D_2) B 2^{-\ell(T_0)/2} \\ &\leq \delta 2^{-\ell(T_0)/2}.\end{aligned}$$

We conduct a further case distinction based on ℓ^*. Consider the case that $\ell^* = 0$. Since we then have $b_0 \geq 1$ and $m(0, j^*) > b_0$ we find $j^* \geq 2$. Hence there exists $1 \leq i \leq j^* - 1$ such that $\ell(T_i) = \ell(T_0)$. Moreover, $T_i \in \mathcal{M}$. The definition of λ now shows

$$\sum_{T' \in \mathcal{M}} \lambda(T_0, T') \geq \lambda(T_0, T_i) = a_{\ell(T_i)-\ell(T_0)} = a_0 > 0$$

Now consider the case that $\ell^* > 0$. For any $1 \leq i \leq j^* - 1$ with $\ell(T_i) = \ell(T_0) + \ell^*$ we have $T_i \in \mathcal{M}$ and $\ell(T_i) \geq \ell(T_0)$ in particular. So the definition of λ shows

$$\lambda(T_0, T_i) = a_{\ell(T_i)-\ell(T_0)} = a_{\ell^*}$$

Therefore

$$\sum_{T' \in \mathcal{M}} \lambda(T_0, T') \geq m(\ell^*, j^*) a_{\ell^*} \geq b_{\ell^*} a_{\ell^*} \geq \inf_{\ell \geq 1} a_\ell b_\ell = \gamma > 0$$

based on the assumptions (7).

To summarize, for any $T_0 \in \mathcal{T}_g \setminus \mathcal{T}_0$ we have

$$\sum_{T' \in \mathcal{M}} \lambda(T_0, T') \geq \min\{a_{-1}, a_{-1-\zeta}, a_0, \gamma\} := C_{\$} > 0$$

which is the desired lower bound. □

Theorem 6.17. *There exists $\Lambda > 0$, depending only on $\mathcal{T}_0$, such that for all sequences $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_g$ of sets of triangles marked for refinement we have*

$$|\mathcal{T}_g| - |\mathcal{T}_0| \leq \Lambda \sum_{i=0}^{g-1} |\mathcal{M}_i|. \tag{13}$$

where

$$\Lambda \leq \frac{C_{\pi,\delta} \left(1 + 2^{1/2+\zeta/2} + (1 + C_{\mathrm{L}} C_{\Sigma}) B\right)^2 A}{\min\{a_{-1}, a_{-1-\zeta}, a_0, \gamma\}} \cdot \frac{D_2^2}{D_1^2}.$$

Proof. This follows from Theorem 6.16 and unfolding definitions. We have previously seen that

$$C_{\mathrm{D}} \leq 1 + C_{\mathrm{L}} C_{\Sigma}, \quad C_{\sharp,\delta} \leq C_{\pi,\delta} D_1^{-2} \left(2^{1/2+\zeta/2} \sqrt{3/4} D_2 + \delta + D_2\right)^2.$$

Hence $\delta = (1 + C_{\mathrm{L}} C_{\Sigma}) D_2 B$ and we bound $C_{\sharp,\delta}$ explicitly by

$$\begin{aligned} C_{\sharp,\delta} &\leq C_{\pi,\delta} D_1^{-2} \left(2^{1/2+\zeta/2} \sqrt{3/4} D_2 + (1 + C_{\mathrm{L}} C_{\Sigma}) D_2 B + D_2\right)^2. \\ &\leq C_{\pi,\delta} D_2^2 D_1^{-2} \left(1 + 2^{1/2+\zeta/2} \sqrt{3/4} + (1 + C_{\mathrm{L}} C_{\Sigma}) B\right)^2. \end{aligned}$$

We come to the desired conclusion. □

Remark 6.18. *The techniques in the proof of Theorem 6.16 resemble the accounting method for amortized analysis in theoretical computer science. The idea of the proof is that every marked triangle deposits a bounded amount of money into the triangles of the last triangulation such that every one of those triangles receives a minimum amount of money. If the total amount of money invested remains non-negative, then the amortized costs per bisection are uniformly bounded.*

The preceding result reproduces the basic proof techniques of prior works. The estimate will deteriorate if the mesh is very non-uniform, that is, the ratio D_2/D_1 is very large. This dependence on the geometry appears unnecessary for a combinatorial algorithm. Our recursive newest vertex bisection only depends on the initial triangulation, the initial assignment of refinement edges, and the marking of triangles; it does not depend on the geometry of the triangulation. For that reason, we naturally expect a purely combinatorial amortized complexity estimate, which does not depend on geometric properties of the triangulation. We derive such a combinatorial estimate from the preceding theorem by replacing the initial triangulation by a combinatorially equivalent one which has uniform geometric properties.

For any $T \in \mathcal{T}_0$ we define $\Gamma(0, T) := \{T\}$ and recursively

$$\Gamma(k, T) := \{S \in \mathcal{T}_0 : \exists T' \in \Gamma(k-1, T) : S \cap T' \neq \emptyset\}, \quad k \in \mathbb{N}.$$

The cardinality of $\Gamma(T, k)$ is the number of triangles in a local patch around T. We write $\Gamma(k) := \max_{T \in \mathcal{T}_0} \Gamma(k, T)$.

Theorem 6.19. *There exists $\Lambda > 0$, depending only on $\mathcal{T}_0$, such that for all sequences $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_g$ of sets of triangles marked for refinement we have*

$$|\mathcal{T}_g| - |\mathcal{T}_0| \leq \Lambda \sum_{i=0}^{g-1} |\mathcal{M}_i|, \tag{14}$$

where

$$\Lambda \leq \frac{|\Gamma(\lceil \rho/h \rceil)| \, A \pi \rho^2}{D_1^2 \min\{a_{-1}, a_{-1-\zeta}, a_0, \gamma\}},$$

$$\rho = D_2 + 2^{1/2+\zeta/2}\sqrt{3/4} D_2 + \delta,$$

$$\delta = (1 + C_{\mathrm{L}} C_{\Sigma}) D_2 B, \quad h = \sqrt{3/2}, \quad D_1 = \frac{\sqrt{3}}{2}, \quad D_2 = \sqrt{6},$$

and where $\Gamma > 0$ is such that every triangle from $\mathcal{T}_0$ is sharing a vertex with at most Γ triangles from $\mathcal{T}_0$.

Proof. Let N be the number of vertices $v_1, v_2, \dots, v_N$ in the triangulation $\mathcal{T}_0$. We define a triangulation $\widehat{\mathcal{T}}_0$ embedded in $\mathbb{R}^N$ as follows: whenever $T = [v_i, v_j, v_k] \in \mathcal{T}_0$, then $\widehat{T} = [e_i, e_j, e_k] \in \widehat{\mathcal{T}}_0$. It follows that $\mathcal{T}_0$ and $\widehat{\mathcal{T}}_0$ are combinatorially equivalent and that newest vertex bisection on one triangulation corresponds to newest vertex bisection on the other. To finish the proof, it remains to estimate the relevant parameters that appear in Theorem 6.16.

We analyze the similarity classes of the triangles that appear during refinement. We call these classes type 0, type I, and type II, respectively. Write $L = \sqrt{2}$.

The initial triangulation contains equilateral triangles that we classify as type 0. They have diameter $d_0 = L$, area $A_0 = L^2\sqrt{3}/4$, and shape measure $\mu_0 = 4/\sqrt{3}$. Furthermore, their height is $h = L\sqrt{3}/2$. The first bisection produces right-angled triangles of type I. They have diameter $d_1 = L$, area $A_1 = L^2\sqrt{3}/8$, and shape measure $\mu_1 = 8/\sqrt{3}$. The second bisection produces triangles of type 0 and isosceles triangle of type II. The latter have diameter $d_2 = h = L\sqrt{3}/2$ and area $A_2 = L^2\sqrt{3}/8$, with shape measure $\mu_2 = 2\sqrt{3}$. Triangles of type 0 and II appear only with even level and triangles of type I with odd level. Writing $D_1 = \frac{\sqrt{3}}{2}$ and $D_2 = \sqrt{6}$, we observe the inequalities

$$D_1 2^{-\ell(T)/2} \leq \operatorname{meas}(T)^{\frac{1}{2}} \leq \operatorname{diam}(T) \leq D_2 2^{-\ell(T)/2}.$$

Next, we estimate the constant $C_{\sharp,\delta}$ for some $\delta > 0$. Let $T \in \mathcal{T}$ be a descendant of $T_0 \in \mathcal{T}_0$ and $k \in \mathbb{N}$. We define the set

$$C(T, \delta, k) := \left\{ S \in \mathcal{T} \mid \operatorname{dist}(S, T) \leq \delta 2^{-k/2}, \ell(S) = k \right\}$$

Suppose that $S \in C(T, \delta, k)$ with $1 \leq k \leq \ell(T) + \zeta + 1$. Then S and T are contained in a ball centered at a point in T and of radius bounded by

$$\begin{aligned}
&\sqrt{3/4}\operatorname{diam}(T) + \delta 2^{-k/2} + \operatorname{diam}(S) \\
&\leq \sqrt{3/4} D_2 2^{-\ell(T)/2} + \delta 2^{-k/2} + D_2 2^{-\ell(T)/2} \\
&\leq \sqrt{3/4} D_2 2^{1/2+\zeta/2} 2^{-k/2} + \delta 2^{-k/2} + D_2 2^{-k/2} \\
&\leq \left(D_2 + 2^{1/2+\zeta/2}\sqrt{3/4} D_2 + \delta \right) 2^{-k/2}.
\end{aligned}$$

We write $\rho := D_2 + 2^{1/2+\zeta/2}\sqrt{3/4}D_2 + \delta$. The ball of radius $\rho 2^{-k/2}$ around any point in $T_0 \in \mathcal{T}_0$ is contained within triangles of the set $\Gamma\left(\lceil \rho 2^{-k/2}/h \rceil, T_0\right)$.

So the area of that ball is at most $\left|\Gamma\left(\lceil \rho 2^{-k/2}/h \rceil, T_0\right)\right| \pi\rho^2 2^{-k}$. Every triangle in $C(T,k)$ has area at least $D_1^2 2^{-k}$. Consequently,

$$|C(T,\delta,k)| \leq \left|\Gamma\left(\lceil \rho 2^{-k/2}/h \rceil, T_0\right)\right| \pi\rho^2/D_1^2.$$

This yields an upper bound for $C_{\sharp,\delta}$.

Letting $\delta = (C_{\mathrm{D}} + D_2)B = (1 + C_{\mathrm{L}}C_\Sigma)BD_2$, Theorem 6.16 gives

$$\Lambda \leq \frac{C_{\sharp,\delta}A}{\min\{a_{-1}, a_{-1-\zeta}, a_0, \gamma\}} \leq \frac{\left|\Gamma\left(\lceil \rho/h \rceil\right)\right| A\pi\rho^2}{D_1 \min\{a_{-1}, a_{-1-\zeta}, a_0, \gamma\}}.$$

The proof is complete. □

Remark 6.20. *The difference between Theorem 6.16/Corollary 6.17 and Theorem 6.19 becomes relevant with very non-uniform meshes, such as the procedurally generated triangulation in Figure 5. Our combinatorial result is uniform for further levels of refinement. The only triangulation properties that enter the combinatorial estimate are C_{L} and $\Gamma(\cdot)$. In applications C_{L} is small and $\Gamma(\cdot)$ can be calculated easily in linear time complexity.*

Remark 6.21. *In many applications marked triangles are to be refined by b generations for some $b \geq 1$. We can incorporate this into our framework with additional markings. The constant in the amortized complexity estimate is then $\left(2^b - 1\right)\Lambda$.*

7. Initial Assignment of Refinement Edges

Our refinement method requires an acyclic assignment of refinement edges in the initial triangulation. While producing such an acyclic assignment is not particularly difficult as such, there are still open questions with regards to the computational details of this problem. We review a number of algorithms for assigning initial refinement edges, distinguishing two classes: algorithms that retain the initial triangles and algorithms that perform refinement in a preprocessing step.

Let us first review assignment procedures that do not modify the initial triangulation. A completely arbitrary initial choice of refinement edges may produce circular dependencies, so we need to take care ensuring acyclicity of the initial refinement.

We can find an acyclic initial assignment of refinement edges with an algorithm that runs in linear time and parallelizes. Suppose that we fix a total order of the edges in the initial triangulation. For example, we can order the edges lexicographically by their vertex indices. If we assign to each triangle its edge that is maximal with respect to that total order, then the resulting assignment will be acyclic. However, the resulting assignment may feature long initial refinement chains.

We can produce an initial assignment of refinement edges in linear serial time such the length of all dependency chains is at most 1; see Algorithm 5. We first greedily assign refinement edges such that only compatibly bisectable pairs are produced, and then assign refinement edges to any remaining triangles such all dependency chains have length at most one.

Finally, if the initial triangulation triangulates a planar domain, there exists an algorithm running in linear serial time that produces an initial assignment of refinement edges such that the triangulation will be ideally matched [5]. This is based

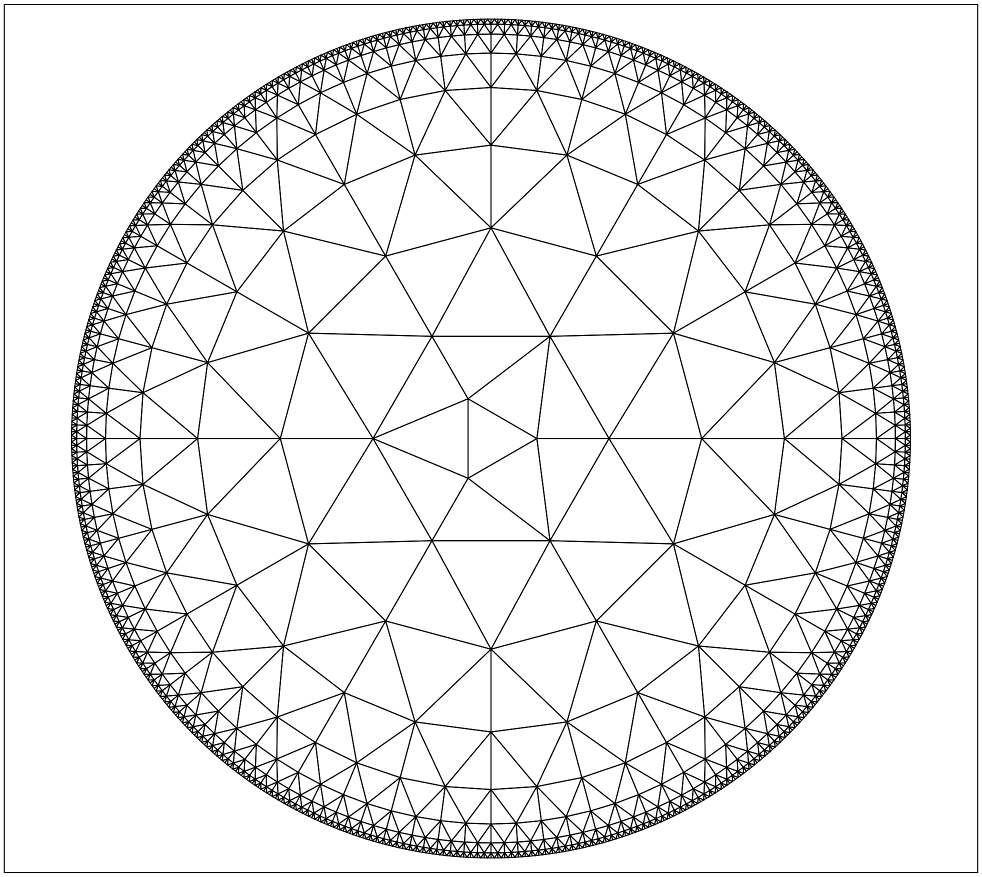


FIGURE 5. Procedurally generated triangulation approximating a disk. The quotient of maximum triangle diameter and minimum triangle area diverges to infinity as the resolution is increased.

Algorithm 5 Greedy Initial Assignment Algorithm

procedure GREEDYINITIALASSIGNMENT($\mathcal{T}$)
    Set $\mathcal{S} = \emptyset$.
    while $\exists e \in \mathcal{E}(\mathcal{T})$: both parent triangles $T_1, T_2 \in \mathcal{T}$ of e are unassigned **do**
        Set $R(T_1) = e$ and $R(T_2) = e$.
        Set $\mathcal{S} := \mathcal{S} \cup \{T_1, T_2\}$.
    while $\exists T \in \mathcal{T}$ unassigned **do**
        Set $R(T) = e$ for some edge e of T such that $\exists S \in \mathcal{S} : e \subset S$.

on reducing the problem to finding a perfect matching in a 3-regular planar graph. It is not obvious how this algorithm can be parallelized. It also seems to be an open question how to achieve such an optimal complexity bound for more general manifold-like triangulations.

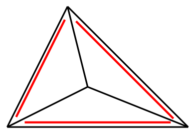
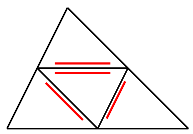
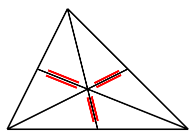
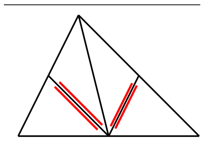
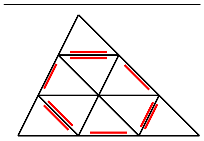
Refinement	Notes
	Midpoint refinement. Smallest cardinality. All bisectable pairs across triangle boundaries.
	Uniform refinement. Original triangle shapes preserved. Dependencies of depth at most one, all within original triangle.
	Refined midpoint refinement. Triangle shapes better than midpoint refinement, all dependencies within original triangle.
	Repeated bisection. All dependencies within original triangle. Non-uniform across triangulation.
	3-uniform refinement. Original shapes preserved. All triangles compatibly bisectable. Dependencies across original triangles.

Table 1. Some Schemes for initial refinement with additional notes.

The second class of algorithms for the initial assignment of refinement edges refines the initial mesh in a preprocessing step. These procedures are comparatively simple, have linear time complexity, are highly parallelizable, and produce assignments with dependency chains at length at most one. That being said, they typically increase the cardinality of the triangulation by a constant factor. While this may be deemed tolerable in some applications (such as finite element methods), it may be deemed intolerable in others (such as boundary element methods or resource critical settings). Table 1 gives a graphical description of some ways to refine the initial triangles and assign refinement edges in that new triangulation.

References

[1] Arnold, D.N., Mukherjee, A., Pouly, L.: Locally adapted tetrahedral meshes using bisection. SIAM Journal on Scientific Computing **22**(2), 431–448 (2000)

[2] Atalay, F.B., Mount, D.M.: Bounds on the cost of compatible refinement of simplex decomposition trees in arbitrary dimensions. Computational Geometry **79**, 14–29 (2019)

[3] Bänsch, E.: Local mesh refinement in 2 and 3 dimensions. IMPACT of Computing in Science and Engineering **3**(3), 181–191 (1991)

[4] Bartels, S., Schreier, P.: Local coarsening of simplicial finite element meshes generated by bisections. BIT **52**(3), 559–569 (2012). DOI 10.1007/s10543-012-0378-0

[5] Biedl, T.C., Bose, P., Demaine, E.D., Lubiw, A.: Efficient algorithms for Petersen's matching theorem. Journal of Algorithms **38**(1), 110–134 (2001)

[6] Binev, P., Dahmen, W., DeVore, R.: Adaptive finite element methods with convergence rates. Numerische Mathematik **97**(2), 219–268 (2004)

[7] Bonito, A., Cascón, J.M., Mekchay, K., Morin, P., Nochetto, R.H.: High-order AFEM for the Laplace–Beltrami operator: Convergence rates. Foundations of Computational Mathematics **16**(6), 1473–1539 (2016)
[8] Bonito, A., Cascón, J.M., Morin, P., Nochetto, R.H.: AFEM for geometric PDE: the Laplace–Beltrami operator. In: Analysis and numerics of partial differential equations, pp. 257–306. Springer (2013)
[9] Bonito, A., Demlow, A.: A posteriori error estimates for the Laplace–Beltrami operator on parametric C^2 surfaces. SIAM Journal on Numerical Analysis **57**(3), 973–996 (2019)
[10] Bonito, A., Demlow, A., Owen, J.: A priori error estimates for finite element approximations to eigenvalues and eigenfunctions of the Laplace–Beltrami operator. SIAM Journal on Numerical Analysis **56**(5), 2963–2988 (2018)
[11] Burman, E., Hansbo, P., Larson, M.G., Larsson, K., Massing, A.: Finite element approximation of the Laplace–Beltrami operator on a surface with boundary. Numerische Mathematik **141**(1), 141–172 (2019)
[12] Camacho, F., Demlow, A.: L^2 and pointwise a posteriori error estimates for FEM for elliptic PDEs on surfaces. IMA Journal of Numerical Analysis **35**(3), 1199–1227 (2015)
[13] Chen, L., Zhang, C.: A coarsening algorithm on adaptive grids by newest vertex bisection and its applications. Journal of Computational Mathematics **28**(6), 767–789 (2010)
[14] Demlow, A.: Higher-order finite element methods and pointwise error estimates for elliptic problems on surfaces. SIAM Journal on Numerical Analysis **47**(2), 805–827 (2009)
[15] Demlow, A.: Quasi-optimality of adaptive finite element methods for controlling local energy errors. Numerische Mathematik **134**(1), 27–60 (2016)
[16] Demlow, A., Dziuk, G.: An adaptive finite element method for the Laplace–Beltrami operator on implicitly defined surfaces. SIAM Journal on Numerical Analysis **45**(1), 421–442 (2007)
[17] Demlow, A., Stevenson, R.: Convergence and quasi-optimality of an adaptive finite element method for controlling L^2 errors. Numerische Mathematik **117**(2), 185–218 (2011)
[18] Dziuk, G.: Finite elements for the Beltrami operator on arbitrary surfaces. In: Partial differential equations and calculus of variations, pp. 142–155. Springer (1988)
[19] Dziuk, G., Elliott, C.: L^2-estimates for the evolving surface finite element method. Mathematics of Computation **82**(281), 1–24 (2013)
[20] Dziuk, G., Elliott, C.M.: Finite element methods for surface pdes. Acta Numerica **22**, 289–396 (2013)
[21] Formaggia, L., Fumagalli, A., Scotti, A., Ruffo, P.: A reduced model for Darcy's problem in networks of fractures. ESAIM: Mathematical Modelling and Numerical Analysis **48**(4), 1089–1116 (2014)
[22] Holst, M.: Adaptive numerical treatment of elliptic systems on manifolds. Advances in Computational Mathematics **15**(1-4), 139–191 (2001)
[23] Holst, M., Stern, A.: Geometric variational crimes: Hilbert complexes, finite element exterior calculus, and problems on hypersurfaces. Foundations of Computational Mathematics **12**(3), 263–293 (2012)
[24] Karkulik, M., Pavlicek, D., Praetorius, D.: On 2d newest vertex bisection: optimality of mesh-closure and h 1-stability of l 2-projection. Constructive Approximation **38**(2), 213–234 (2013)
[25] Kossaczkỳ, I.: A recursive approach to local mesh refinement in two and three dimensions. Journal of Computational and Applied Mathematics **55**(3), 275–288 (1994)
[26] Liu, A., Joe, B.: Quality local refinement of tetrahedral meshes based on bisection. SIAM Journal on Scientific Computing **16**(6), 1269–1291 (1995)
[27] Maubach, J.M.: Local bisection refinement for n-simplicial grids generated by reflection. SIAM Journal on Scientific Computing **16**(1), 210–227 (1995)
[28] Mekchay, K., Morin, P., Nochetto, R.: AFEM for the Laplace–Beltrami operator on graphs: design and conditional contraction property. Mathematics of Computation **80**(274), 625–648 (2011)
[29] Mitchell, W.F.: Unified multilevel adaptive finite element methods for elliptic problems. Ph.D. thesis, Citeseer (1988)
[30] Nochetto, R.H., Siebert, K.G., Veeser, A.: Theory of adaptive finite element methods: an introduction. In: Multiscale, nonlinear and adaptive approximation, pp. 409–542. Springer (2009)

[31] Nochetto, R.H., Veeser, A.: Primer of adaptive finite element methods. In: Multiscale and adaptivity: modeling, numerics and applications, pp. 125–225. Springer (2011)
[32] Nordbotten, J.M., Boon, W.M., Fumagalli, A., Keilegavlen, E.: Unified approach to discretization of flow in fractured porous media. Computational Geosciences **23**(2), 225–237 (2019)
[33] Olshanskii, M.A., Reusken, A., Grande, J.: A finite element method for elliptic equations on surfaces. SIAM journal on numerical analysis **47**(5), 3339–3358 (2009)
[34] Rivara, M.C.: Selective refinement/derefinement algorithms for sequences of nested triangulations. International journal for numerical methods in Engineering **28**(12), 2889–2906 (1989)
[35] Rivara, M.C.: Local modification of meshes for adaptive and/or multigrid finite-element methods. Journal of Computational and Applied Mathematics **36**(1), 79–89 (1991)
[36] Stevenson, R.: The completion of locally refined simplicial partitions created by bisection. Mathematics of computation **77**(261), 227–241 (2008)
[37] Traxler, C.T.: An algorithm for adaptive mesh refinement in n dimensions. Computing **59**(2), 115–137 (1997)

UCSD Department of Mathematics, 9500 Gilman Drive MC0112, La Jolla, CA 92093-0112
Email address: `mholst@ucsd.edu`

UCSD Department of Mathematics, 9500 Gilman Drive MC0112, La Jolla, CA 92093-0112
Email address: `mlicht@ucsd.edu`

Harvard John A. Paulson School of Engineering and Applied Sciences, 29 Oxford Street, Cambridge, MA 02138
Email address: `zlyu@g.harvard.edu`