Simplicial Mesh Refinement in Computational Geometry

Zhao Lyu

Department of Mathematics, University of California, San Diego

Contents

1 Introduction

One question in computational geometry is how to represent complicated geometric shapes as a composition of more simple shapes. For example, an area can be expressed as a composition of triangles, and a volume can be written as a composition of tetrahedra. In this thesis, we discuss an important topic about such decompositions, namely the refinement of simplicial meshes.

One motivation for this topic is the numerical analysis of partial differential equations. While numerical methods simulate the behavior of a partial differential equation over some area, it is necessary to represent the area on a computer first. This process is called triangulation, decomposing the area into simplices. Importantly, we sometimes may want to increase the resolution of this triangulation. To do so, algorithms which perform simplicial mesh refinement are crucial and worthy of attention. Besides its application in partial differential equations [7], mesh refinement can also be applied in other fields, such as solving interpolation problems [13].

There are two main challenges in designing such algorithms. One is to maintain the stability of these simplices. In other words, unlimited to how many times a refinement is repeated, shapes of triangles should be bounded. That is, there should not exist any degenerating triangles, which refer to those with extremely small angles. One main reason why degenerating triangles should be avoided is that these triangles lead to ill-conditioned matrices in numerical methods for partial differential equations [2]. Another challenge is to preserve the consistency of the triangulation. This means that two triangles either do not touch or only touch at a common edge or vertex, and the importance of this attribute is that an algorithm is expected to refine triangles consistently and successfully. In short, these two constraints make the development of algorithms for simplicial mesh refinement a challenging problem.

In this thesis, we discuss two applicable algorithms for mesh refinement in two dimensions. The first algorithm is called uniform refinement, one popular global refinement, of which the refinement is done all at once [3, 4, 5]. Though its easy application and obvious qualification for stability and consistency bring a popularity to this refinement strategy, it fails to provide some flexibility in refining simplicial meshes since uniform refinement is applied to whole simplices simultaneously. This is because uniform refinement forces to refine the entire mesh at once. However, many applications demand the flexibility to refine the mesh only locally. Thus, another algorithm, called the newest vertex bisection, is introduced to obtain such flexibility for local mesh refinement. However, newest vertex bisection is not perfect either. While newest vertex bisection of a single triangle preserves stability, preservation of consistency becomes complicated. In detail, the problem is that bisecting one triangle may depend on bisection of another. This leads to a chain reaction, which can be understood as a recursion using a stack and traversing back only when a base case is touched. Therefore, the difficulty is to determine how many triangles are part of this chain reaction, and under which conditions this chain reaction ends. In other words, it is not trivial whether the algorithm terminates and how long it lasts.

In conclusion, this thesis will introduce uniform refinement and newest vertex bisection in two dimensions for computational geometry, and prove their stability and consistency in application, and further discuss their potential limits, possible solutions and their application in three dimensions and difficulties.

2 Basic Definition

In this section, we introduce basic notions which we will use to present and analyze mesh refinement. We first recall definitions of linear transformation and translation, and further introduce affine transformation.

2.1 Translation, Linear Transformation and Affine Transformation

We define several classes of transformations that we frequently use.

Definition 1. Let $v \in \mathbb{R}^n$. A *translation* T_v is a mapping of the form $T_v(x) = x + v$, for any vector $x \in \mathbb{R}^n$.

A translation moves every point of a figure or space by the same distance in the same direction. A translation T can be represented by an addition of a constant vector to every point.

Definition 2. We say a function $f : \mathbb{R}^n \to \mathbb{R}^n$ is a *linear transformation* if the following is satisfied:

$$\begin{aligned} f(u+v) &= f(u) + f(v) \qquad \forall u, v \in \mathbb{R}^n, \\ f(cu) &= cf(u), \qquad \forall u \in \mathbb{R}^n,\ c \in \mathbb{R}. \end{aligned}$$

In other words, a linear transformation is a mapping which preserves the operations of vector addition and scalar multiplication. We can represent the linear transformation f by a matrix M. For example, if M is an $m \times n$ matrix, then f is a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Definition 3. An *affine transformation* from $\mathbb{R}^n$ to $\mathbb{R}^n$ is of the form

$$F(x) = Ax + v, \qquad x \in \mathbb{R}^n,$$

where $A \in \mathbb{R}^{n \times n}$ is a matrix, and $v \in \mathbb{R}^n$ is a vector.

The inverse mapping of an affine transformation $F(x) = Ax + v$ is only defined if A^{-1} exists, and then the inverse mapping $F^{-1}(x) = x \mapsto A^{-1}(x - v)$ is also an affine transformation. Affine transformation preserves points, lines and planes, but need not preserve the origin in a linear space in contrast to the linear transformation. So we see that translations and linear transformations are affine, but the opposite is not true. Affine transformations help carry results from one simplex to another simplex in our discussion, and more details are covered after introducing *simplices* and *triangulations* in the next section.

2.2 Simplices

In this section, we will introduce simplices and talk about its geometrical properties, such as diameter and volume. These information and notation are mainly from [6][5].

Definition 4. A k-simplex $T \subseteq \mathbb{R}^n$ is a convex hull of $k+1$ vertices $x_0, \ldots, x_k \in \mathbb{R}^n$, which are affinely independent. We write

$$\begin{aligned} T &:= [x_0, \cdots, x_k] \\ &:= \left\{ x = \sum_{i=0}^{k} \lambda_i x_i \;\middle|\; \sum_{i=0}^{k} \lambda_i = 1 \text{ and } 0 \leqslant \lambda_i \leqslant 1, 0 \leqslant i \leqslant k \right\} \\ &:= \left\{ \lambda_0 x_0 + \cdots + \lambda_k x_k \;\middle|\; \sum_{i=0}^{k} \lambda_i = 1 \text{ and } 0 \leqslant \lambda_i \leqslant 1, 0 \leqslant i \leqslant k \right\}. \end{aligned}$$

If $k = n$, we do not address the dimension of a k-simplex. 2-simplices are also called *triangles*, and 3-simplices are called *tetrahedra*.

Definition 5. An l-simplex $S = [y_0, \cdots, y_l]$ is called an *l-subsimplex* of a k-simplex $T = [x_0, \cdots, x_k]$, if there exist indices $0 \leqslant i_0 < \ldots < i_l \leqslant k$ with $y_{i_j} = x_i$, for all $0 \leqslant j \leqslant l$.

Since there are $k+1$ vertices in a k-simplex T, and $l+1$ vertices in l-subsimplices S, the number of l-subsimplices of k-simplices is $\binom{k+1}{l+1}$.

Simplices under Affine Transformation
Let F be an affine transformation. Instead of taking a single variable $x \in \mathbb{R}^n$ for affine transformation, we can take a subset S $\subseteq \mathbb{R}^n$, which contains $x \in \mathbb{R}^n$. Then the transformed set S' is

$$S' = \{F(x) \mid x \in S\}.$$

Similarly, if we regard a k-dimensional simplex $T = [x_0, \cdots, x_k]$ as a subset of $\mathbb{R}^n$ then the image of T under affine transformation is

$$F(T) = [F(x_0), \cdots, F(x_k)].$$

We can see that $F(T)$ is still a k-dimensional simplex. Let us write $T' = F(T)$. We might be curious about the relationship between simplices T and T'. An important property of simplices is congruence.

Definition 6. Two simplices T, T' are defined to be congruent if they can be obtained from each other by rotation, mirroring, scaling, and translation, i.e. if there exists a scaling factor c $\in \mathbb{R}^+$, a translation vector $v \in \mathbb{R}^n$, and an orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ such that

$$T' = v + cQT.$$

Then we say that T and T' are in a same congruence class. Formally, a congruent class is an equivalence class of simplices under the congruence relation. We write $T \cong T'$ if T and T' are simplices in the same congruence class. When T and T' share same shape but not necessarily same size, we say T is similar to T', and write $T \sim T'$.

2.3 Shape Regularity Measure

Shape measure offers an objective mathematical measure on the overall quality of a simplex, and this is helpful to explore the simplex regularity and to improve the quality of shapes of the elements.

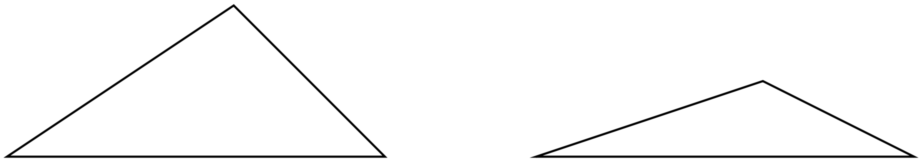

Figure 1: Good triangle(left) with smaller shape measure vs.
Bad triangle(right) with larger shape measure.

Different definitions are used for shape measure to present the quality of simplex, and we simply introduce the *geometric shape measure* $\mu(T)$ of simplex T, the one we use in this paper.

Simplex Diameter and Volume
Let $T \in \mathbb{R}^n$ be a k-simplex where $k \leqslant n$, with vertices $x_0, \cdots, x_k \in \mathbb{R}^n$. We let $\mathrm{diam}(T)$ denote the diameter of T, and we see

$$\mathrm{diam}(T) = \max_{0 \leqslant i \leqslant j \leqslant k} \|x_i - x_j\|.$$

In other words, $\mathrm{diam}(T)$ is the longest distance between two vertices of T, which is the length of the longest edge of T. If T is a single vertex, then $\mathrm{diam}(T) = 0$.

Let $\mathrm{vol}^k(T)$ denote the k-dimensional volume of T. We have

$$\mathrm{vol}^{\mathrm{k}}(T) = \frac{1}{k!} \cdot |\det(x_1 - x_0, x_2 - x_0, \cdots, x_k - x_0)|.$$

If k = 0, then T is a 0-dimensional simplex, i.e., a vertex. By convention we have $\mathrm{vol}^0(T) = 1$, which means that the volume of a single vertex is one.

Shape Measure
Simplex diameter and volume are important to introduce shape measures. Here we define the *shape measure* $\mu(T)$ of a k-simplex T by

$$\mu(T) = \frac{\mathrm{diam}(T)^k}{\mathrm{vol}^{\mathrm{k}}(T)}, \quad \mathrm{vol}^{\mathrm{k}}(\mathrm{T}) \neq 0\,.$$

If $\mathrm{vol}^k(T) = 0$, then we define $\mu(T) = \infty$.

To understand this definition, we can interpret $\mu(T)$ as a measurement of how different the two variables $\mathrm{diam}(T)^k$ and $\mathrm{vol}^k(T)$ are. For example, for a 2-dimensional simplex, i.e. a triangle, shape measure helps measure how narrow the triangle is. In other words, it measures how small the smallest angle of the triangle is. A triangle with a fixed diameter whose smallest angle gets smaller and smaller will look more and more like a one-dimensional line, and its shape measure will diverge to infinity.

Lemma 1. If T and T' are congruent simplices, then $\mu(T) = \mu(T')$.

Proof. Since T is congruent to T', by definition, we have $T' = v + cQT$, where $c \in \mathbb{R}^+$ is scaling factor, $v \in \mathbb{R}^n$ is a translation vector and $Q \in O(n)$ is an orthogonal matrix. In fact, we will show that scalings, translations, orthogonal transformation do not influence the shape measure of a simplex.

To be specific, when scaling a simplex T by a non zero factor $c \in \mathbb{R}^+$ to obtain T', we have

$$\begin{aligned}\mathrm{vol}^k(T') &= \frac{1}{k!} \cdot |\det(cx_1 - cx_0, cx_2 - cx_0, \cdots, cx_k - cx_0)| \\ &= \frac{c^k}{k!} \cdot |\det(x_1 - x_0, x_2 - x_0, \cdots, x_k - x_0)| = c^k \cdot \mathrm{vol}^k(T).\end{aligned}$$

Since it scales over all vertices, $\mathrm{diam}(T')^k = c^k \cdot \mathrm{diam}(T)^k$. Therefore, we see

$$\mu(T') = \frac{\mathrm{diam}(T')^k}{\mathrm{vol}^k(T')} = \frac{c^k \cdot \mathrm{diam}(T)^k}{c^k \cdot \mathrm{vol}^k(T)} = \frac{\mathrm{diam}(T)^k}{\mathrm{vol}^k(T)} = \mu(T).$$

Moreover, translation of a simplex T by a nonsingular vector v to obtain T' will not influence the shape measure as well. In detail, we have

$$\begin{aligned}\mathrm{diam}(T')^k &= \max_{0 \leqslant i \leqslant j \leqslant k} \|(x_i + v) - (x_j + v)\| \\ &= \max_{0 \leqslant i \leqslant j \leqslant k} \|x_i - x_j\| = \mathrm{diam}(T)^k\end{aligned}$$

and

$$\begin{aligned}\mathrm{vol}^k(T') &= \frac{1}{k!} \cdot |\det((x_1 + v) - (x_0 + v), (x_2 + v) - (x_0 + v), \cdots, (x_k + v) - (x_0 + v))| \\ &= \frac{1}{k!} \cdot |\det(x_1 - x_0, x_2 - x_0, \cdots, x_k - x_0)| = \mathrm{vol}^k(T),\end{aligned}$$

so that again

$$\mu(T') = \frac{\mathrm{diam}(T')^k}{\mathrm{vol}^k(T')} = \frac{\mathrm{diam}(T)^k}{\mathrm{vol}^k(T)} = \mu(T).$$

Consider rotating and mirroring T by an orthogonal matrix Q to obtain T'. Since multiplying a vector with Q does not change its length, we have

$$\mathrm{diam}(T') = \max_{0 \leqslant i \leqslant j \leqslant k} \|Qx_i - Qx_j\| = \max_{0 \leqslant i \leqslant j \leqslant k} \|x_i - x_j\| = \mathrm{diam}(T)$$

and since $|\det Q| = 1$, we have

$$\begin{aligned}\mathrm{vol}^k(T') = \mathrm{vol}^k(Q \cdot T) &= \frac{1}{k!} \cdot |\det(Q(x_1 - x_0), Q(x_2 - x_0), \cdots, Q(x_k - x_0)| \\ &= \frac{1}{k!} \cdot |\det(Q)| \cdot |\det(x_1 - x_0, x_2 - x_0, \cdots, x_k - x_0)| \\ &= \frac{1}{k!} \cdot |\det(x_1 - x_0, x_2 - x_0, \cdots, x_k - x_0)| = \mathrm{vol}^k(T).\end{aligned}$$

Therefore, we obtain

$$\mu(T') = \frac{\mathrm{diam}(T')^k}{\mathrm{vol}^k(T')} = \frac{\mathrm{diam}(T)^k}{\mathrm{vol}^k(T)} = \mu(T).$$

Now we see that the shape measure is independent of scaling, translation, rotation or mirroring. Thus a simplex T' which is obtained by these motions shares a same shape measure with T. □

The reason why we need the notion of shape measure is to help to understand whether a simplex T is non-degenerate, and to quantify how degenerate or non-degenerate. Let T be a k-dimensional simplex in $\mathbb{R}^n$. We say that a simplex T is degenerate if $\mu(T) = \infty$, i.e. $\mathrm{vol}^k(T) = 0$.

Observing two triangles in 1, we actually want the interior angles of the simplex T, i.e. triangles in this example, to be uniformly bounded from zero. While cutting a simplex into smaller pieces, we want to keep the shape measures of the simplices uniformly bounded and avoid degenerate simplices.

2.4 Simplicial Complexes

Definition 7. A *simplicial complex* $\mathcal{T}$ in $\mathbb{R}^n$ is a finite set of simplices in $\mathbb{R}^n$ that satisfies the following conditions:

1. Any subsimplex of a simplex from $\mathcal{T}$ is also in $\mathcal{T}$.
2. The intersection of any two simplices $T_1, T_2 \in \mathcal{T}$ is a face of both T_1 and T_2.

In other words, the first condition asks $\mathcal{T}$ to be closed under taking subsimplices, and the second condition asks that the intersection of any two simplices is either a common subsimplex or empty because the empty set is a face of every simplex. Examples in 2D are shown in Figure 2.

Any subset $T' \in T$ that is itself a simplicial complex is called a *subcomplex* of T. We say that a *simplicial k-complex* $\mathcal{T}$ is a simplicial complex where the largest dimension of any simplex in $\mathcal{T}$ is k. So a simplicial 2-complex must not contain tetrahedra or higher dimension simplices. The 0-complex of T is called a *vertex set* of T. We can also think of a simplicial complex as a space with a triangulation, which is the division of a surface or a plane polygon into a set of 2-simplices.

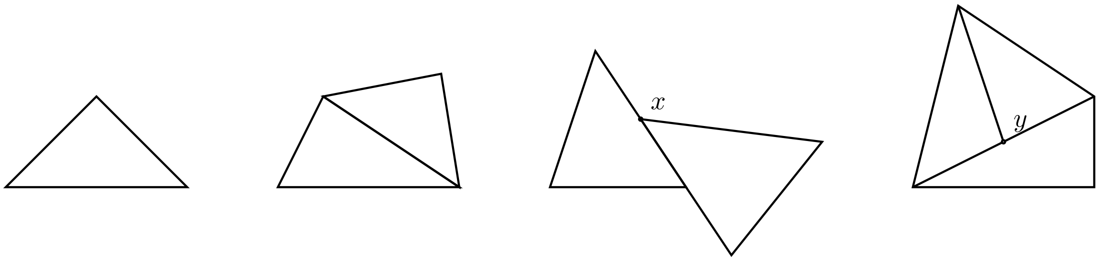

Figure 2: Simplicial complex(left); Not simplicial complex(right)
The two arrangements of simplices on the right are not simplicial complexes because their intersection x and y are not shared. We call such nodes like x and y hanging node.

Shape Measure of Simplicial Complex
Recall the definition of the shape measure of a simplex. Now consider a simplicial complex $\mathcal{T}$, we define the geometric shape measure $\mu(\mathcal{T})$ as follows,

$$\mu(\mathcal{T}) := \max_{T \in \mathcal{T}} \mu(T).$$

By definition, we see that the shape measure of a simplicial complex $\mathcal{T}$ is the supremum of the set of shape measures of all simplices $T \in \mathcal{T}$. If the largest shape measure of a simplex in this simplicial complex is bounded, then none of the simplices in $\mathcal{T}$ are degenerate.

3 Refinement Strategies in General

Suppose that a domain is divided into simplices. Mesh refinement is a procedure of mesh modification in which we divide these simplices into smaller simplices. This process can be applied recursively. Let us first introduce triangulation to help understand refinement on a simplex. Generally speaking, we can think triangulation as a subdivision of a plane into triangles. The following definition is a more formal way to take when extending to a higher dimension.

Definition 8. A triangulation of $\mathbb{R}^n$ is subdivision into n-dimensional simplices such that intersection of any two simplices is either empty or sharing a common face, and any face of a simplex is in the triangulation.

Indeed, we say that this triangulation is consistent as it is not simply subdividing of a space. Moreover, the triangulation defined here can be treated equivalently as simplicial complex as it is a finite set of simplices satisfying

1. Any face of a simplex from a triangulation is also in the triangulation
2. The intersection of any two simplices T_1, T_2 in a triangulation is a face of both T_1 and T_2 or empty

We can think a refinement of a simplex T as a triangulation $\mathcal{T}$ which consists of smaller pieces of simplices of the same type of the simplex T. Now consider a refinement of a simplicial complex. Let $\mathcal{T}$ and $\mathcal{T}'$ be two different simplicial complexes covering a same domain Ω. This means that the domain $\Omega = \bigcup(T | T \in \mathcal{T}) = \bigcup(T' | T \in \mathcal{T}')$. We say that $\mathcal{T}'$ is a refinement of $\mathcal{T}$ if each simplex $T \in \mathcal{T}$ is in $\mathcal{T}'$ or the triangulation of T is in $\mathcal{T}'$.

As mentioned before, we may recursively apply a refinement strategy to help simplify some problems. By recursively taking refinement process from $\mathcal{T}_0$, we have a hierarchy triangulation $\mathcal{T}_k, k \in \mathbb{N}$, where $\mathcal{T}_k$ is a refinement of $\mathcal{T}_{k-1}$.

Definition 9. Let $\mathcal{T}_0$ be the initial simplicial complex in $\mathbb{R}^n$ where it starts from, then we define the hierarchy triangulation $\mathcal{T}_k$ as follows

$$\mathcal{T}_k := \bigcup\{refinement\ of\ simplex\ T \mid T \in \mathcal{T}_{k-1}\}, \quad k \in \mathbb{N}.$$

3.1 Consistency of Refinement

We want the triangulation always to be consistent after applying a refinement. This feature is proved in section 2.4 that if either 1) any face of a simplex from this triangulation $\mathcal{T}$ is also in $\mathcal{T}$, or 2) the intersection of any two simplices in a face of both simplices.

3.2 Stability of Refinement

Besides consistency, we also want all simplices in a triangulation resulted from a refinement strategy non-degenerating so that we can apply the refinement strategy recursively to have nicely shaped triangulation in the end.

Definition 10. We say a refinement strategy is **stable** if there exists a constant $C > 0$ such that $\mu(T) < C$ for all simplices T.

Theorem. If the number of congruence classes, obtained by applying the refinement of a non-degenerate simplex T initially, is finite, then the refinement strategy is stable.

Proof. We claim that a refinement strategy over initial simplicial complex T_0 produces only non-degenerate simplices T.

We prove this claim by induction. Clearly, the base case is true since it is given that all simplices T in $\mathcal{T}$ are non-degenerate. For induction, suppose simplices in simplicial complex $\mathcal{T}_k$ is non-degenerate. That is, there exists $C > 0$ such that $\mu(T) < C$, $\forall T \in \mathcal{T}_k$. Apply the refinement strategy on $\mathcal{T}_k$, and then we obtain $\mathcal{T}_{k+1} = \bigcup\{refinement\ of\ simplex\ T \mid T \in \mathcal{T}_k\}, \quad k \in \mathbb{N}$.

Next, we show the following fact. If the number of congruence classes is finite, then the number of shape measure is finite, and there exists a common bound $C > 0$ such that $C \geq \mu(\mathcal{T})$.

This can be seen as follows. We proved that simplices in same congruence classes share the same shape measure. If we have a finite number of congruence classes, clearly we have a finite number of shape measures. When all simplices are non-degenerate, we always have an upper bound for their shape measure $\mu(T)$. With the finite number of shape measures, we may set C as the maximum of all upper bounds of shape measures. And therefore $C \geq \mu(\mathcal{T})$.

Since T_0 is non-degenerate, T_0 is non-degenerate. Moreover, we know there exists a common bound C for all shape measures since the number of congruence classes is finite. Therefore, we proved the stability. □

4 Uniform Refinement

Based on how we preserve stability and consistency, one strategy is *red/green refinement algorithm*, and it has been discussed by Randolph E Bank and other mathematicians[3, 4, 18]. Basically, the red refinement here is regular refinement which divides a triangle into four congruent smaller triangles, and green refinement is necessary to preserve the consistency of the triangulation. *Uniform refinement* generally inherits its idea, but differently, it does not demand a green refinement specifically to help preserve the stability since the refinement is done simultaneously.

4.1 Uniform Refinement Algorithm in 2-dimension

One popular global refinement strategy is *uniform refinement.* The main idea of uniform refinement strategy is to subdivide the triangle into four smaller triangles by connecting midpoints on each edge.

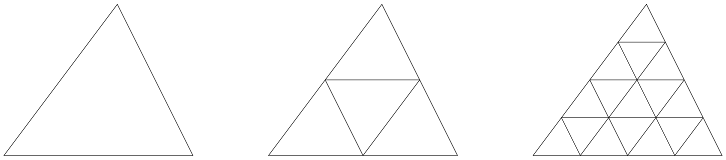

Figure 3: Illustration of uniform refinement
Starting with a single triangle and two successive refinement steps.

Let $T = [x^0, x^1, x^2]$ be the triangle to be refined, and denote x^{ij} by the midpoint of the edge between x^i and x^j for $0 \leq i < j \leq 2$. Uniform refinement of T produces the four new tetrahedra

$$T_1 := [x^0, x^{01}, x^{02}], \qquad T_2 := [x^{01}, x^1, x^{12}],$$
$$T_3 := [x^{02}, x^{12}, x^2], \qquad T_4 := [x^{01}, x^{12}, x^{02}].$$

Lemma 2. The triangle T_1, T_2, T_3 and T_4 produced by the uniform refinement have the same congruence class as T.

Proof. Let A, B, C be the vertices of the triangle T and let X, Y, Z be the midpoints of the edge AB, BC and AC. An application of uniform refinement produces the triangles $T_1 = [A, X, Z]$, $T_2 = [X, B, Y]$, $T_3 = [Z, Y, C]$ and $T_4 = [Y, Z, X]$. See Figure 4.

We have line XY parallel to line AC, i.e., $XY \parallel AC$, so $\angle BXY = \angle XAZ$. Similarily, since $XZ \parallel BC$, we have $\angle XBY = \angle AXZ$. Since X is the midpoint of line AB, then $|AX|= |BX|$. In short, we have

$$\angle BXY = \angle XAZ, \quad |BX|= |AX|, \quad \angle XBY = \angle AXZ.$$

Therefore, $\triangle BXY \cong \triangle XAZ$. Similarily, we can prove the four triangles are congruent to each other, i.e., $\triangle BXY \cong \triangle XAZ \cong \triangle YZC \cong \triangle ZYX$, so they are in a same congruent class, which is same as the one of the original triangle $\triangle BAC$. □

By the lemma 2, we know that uniform refinement applied on a non-degenerate simplex in 2 dimensions gives one congruence class. Moreover, by Theorem in 3.1, we see that the uniform refinement strategy is stable.

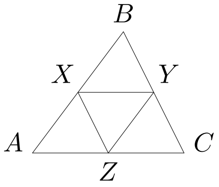

Figure 4: Illustration of Uniform Refinement as in the proof of Lemma 2

Lemma 3. Uniform refinement preserves consistency in the 2-dimensional case.

Clearly, the uniform refinement strategy is stable since it produces a finite number of triangles congruent to the original simplex. Meanwhile, we preserve consistency by bisecting triangles with one refined edge and never refine them any further. Therefore, we obtain stability and consistency through uniform refinement.

4.2 Counting Vertices and Edges by Uniform Refinement

In Figure 4, we see that we obtain four congruent triangles similar to the original large triangle. The lemma 4 below presents relationship between the number of triangles and number of times that uniform refinement is applied.

Lemma 4. Let T_m be the number of triangles after applying uniform refinement m times. Then,

$$T_{m+1} = 4 \cdot T_m, \quad T_m = 4^m \cdot T_0.$$

Proof. Based on Figure 3, we see that we always obtain 4 similar triangles that are in the same congruence class as the original one. In other words, we have $T_{m+1} = 4 \cdot T_m$.

To prove $T_m = 4^m \cdot T_0$ by induction, we have the base case that $T_1 = 4^1 \cdot T_0$ in Figure 3. Suppose that this is true for $T_m = 4^m \cdot T_0$, we need to prove this holds true for T_{m+1}.
Since we have $T_{m+1} = 4 \cdot T_m$, and by inductive hypothesis, we have

$$T_{m+1} = 4 \cdot T_m = 4 \cdot (4^m \cdot T_0) = 4^{m+1} \cdot T_0.$$

Therefore, by induction, we have proved that $T_{m+1} = 4 \cdot T_m$. □

Lemma 5. Let E_m be the number of edges after applying uniform refinement m times. Then,

$$E_{m+1} = 2 \cdot E_m + 3 \cdot T_m, \quad E_m = 2^m \cdot E_0 + 3 \cdot 2^{m-1} \cdot (2^m - 1) \cdot T_0.$$

Proof. After applying the uniform refinement one more time, that is $m + 1$ times in total, then we can think like in each smaller triangle in T_m, we again have a double number of edges by bisecting edges, and obtain the number of T_m edges from connecting each midpoint. Therefore, we have $E_{m+1} = 2 \cdot E_m + 3 \cdot T_m$.

The proof of the second equation can be done by induction. The base case is obvious by Figure 3, that is

$$\begin{aligned} E_1 &= 9 \\ &= 2^1 \cdot 3 + 3 \cdot 2^0 \cdot (2^1 - 1) \cdot 1 \\ &= 2^1 \cdot E_0 + 3 \cdot 2^{1-1} \cdot (2^1 - 1) \cdot T_0. \end{aligned}$$

Suppose that this holds after applying uniform refinement m times, i.e.

$$E_m = 2^m E_0 + 3 \cdot 2^{m-1} \cdot (2^m - 1) \cdot T_0.$$

, since $E_{m+1} = 2 \cdot E_m + 3 \cdot T_m$, and by the inductive hypothesis and lemma 4, we have

$$\begin{aligned}E_{m+1} &= 2 \cdot E_m + 3 \cdot T_m \\ &= 2(2^m \cdot E_0 + 3 \cdot 2^{m-1} \cdot (2^m - 1) \cdot T_0) + 3 \cdot 4^m \cdot T_0 \\ &= 2^{m+1} \cdot E_0 + 3 \cdot 2^m \cdot (2^m - 1) \cdot T_0 + 3 \cdot 4^m \cdot T_0 \\ &= 2^{m+1} \cdot E_0 + 3 \cdot (2^m \cdot (2^m - 1 + 2^m)) \cdot T_0 \\ &= 2^{m+1} \cdot E_0 + 3 \cdot 2^m \cdot (2^{m+1} - 1) \cdot T_0.\end{aligned}$$

Thus, we finished the proof by induction. □

Lemma 6. Denote V_m as the number of vertices after applying uniform refinement m times, then,

$$V_{m+1} = V_m + E_m, \quad V_m = V_0 + (2^m - 1) \cdot E_0 + (2^{m-1} \cdot (2^m + 3) - 2) \cdot T_0.$$

Proof. Whenever uniform refinement is applied, we set a midpoint on each edge as a new vertex. That is, the number of new vertices is the number of edges of the simplicial complex before applying the uniform refinement. Then adding together, we have $V_{m+1} = V_m + E_m$.

We now prove the other equality. By using Lemma lemma 5 repeatedly, we have

$$\begin{aligned}V_{m+1} &= V_m + E_m \\ &= V_m + 2^m \cdot E_0 + 3 \cdot 2^{m-1} \cdot (2^m - 1) \cdot T_0 \\ &= V_0 + \sum_{k=0}^{m} 2^k E_0 + 3 \cdot 2^{k-1} \cdot (2^k - 1) \cdot T_0.\end{aligned}$$

Expanding, $3 \cdot 2^{k-1} \cdot (2^k - 1) = 3 \cdot 2^{k-1+k} + 3 \cdot 2^{k-1}$, we have

$$\begin{aligned}&= V_0 + \sum_{k=0}^{m} 2^k E_0 + 3 \sum_{k=0}^{m} 2^{2k-1} \cdot T_0 + 3 \sum_{k=0}^{m} 2^{k-1} \cdot T_0 \\ &= V_0 + (2^{m+1} - 1)E_0 + \frac{3}{2} \sum_{k=0}^{m} 4^k \cdot T_0 + \frac{3}{2} \sum_{k=0}^{m} 2^k \cdot T_0.\end{aligned}$$

Since $\sum_{k=0}^{m} a^k = \frac{a^{m+1}-1}{a-1}$, we then have

$$\begin{aligned}&= V_0 + (2^{m+1} - 1)E_0 + \frac{3}{2} \cdot \frac{4^{m+1} - 1}{3} \cdot T_0 + \frac{3}{2}(2^{m+1} - 1) \cdot T_0 \\ &= V_0 + (2^{m+1} - 1)E_0 + \frac{4^{m+1} - 1 + 3 \cdot 2^{m+1} - 3}{2} \cdot T_0 \\ &= V_0 + (2^{m+1} - 1)E_0 + (2^m \cdot (2^{m+1} + 3) - 2) \cdot T_0.\end{aligned}$$

Thus, we have finished the proof. □

5 Newest Vertex Bisection

Another popular refinement strategy is bisection. Basically, we cut a triangle by connecting one vertex, which we call the peak, with its opposite edge, which we call refinement edge. Generally, if we apply random bisection with no plan on a triangulation, it is likely that we fail to preserve stability and consistency.

Hence, one important part we need to consider is how to choose the peak for a triangle to preserve the stability and consistency, and one famous method was introduced as the *newest vertex bisection.* In newest vertex bisection, we create the newest vertex at the middle of the refinement edge after applying the bisection refinement once, and then we regard the newest vertex as the peak for bisection over the resulting two smaller triangles.

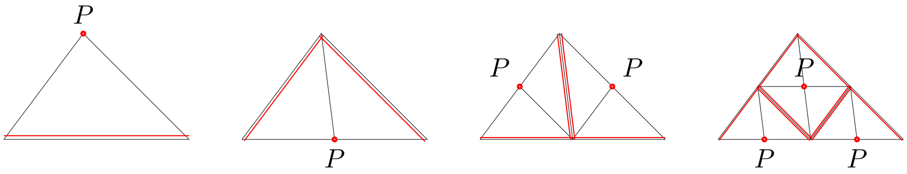

Figure 5: Illustration of bisection refinement, starting with a single triangle
P represents a peak, and red line denotes a refinement edge.

Lemma 7. Bisection refinement gives four congruence classes given one triangle.

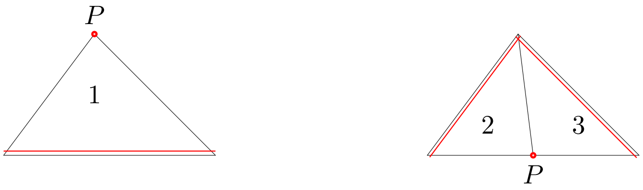

Figure 6: Stage 1: Original triangle(left) in congruence class 1
Stage 2: Applied the newest vertex bisection once(right) in congruence class 2 and 3.

Proof. Observing Figure 6, we have a triangle at the beginning, say in congruency class 1. After applying the newest vertex bisection refinement once, see Figure 6, we obtain two smaller triangles as in the second picture in Figure 5. Say one of them is in the congruency class 2, and another one is in the congruency class 3. Further applying the newest vertex bisection refinement, we have the triangulation in Figure 7.

Claim 1. The left and right bottom triangles in Stage 3 are congruent to the original triangle in Stage 1, and they are in the congruency class 1. Moreover, the other two triangles left are congruent and in the congruency class 4.

Proof of Claim 1:
Let A, B, C be the vertices of the triangle T and let X, Y, Z be the midpoints of the edge AB, AC and BC. An application of the newest vertex bisection refinement produces the triangle

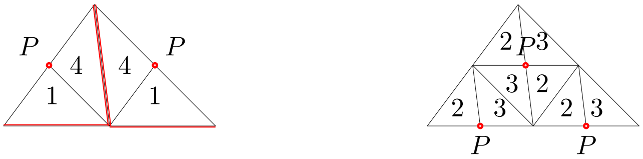

Figure 7: Stage 3: Applied the newest vertex bisection twice(left) in congruence class 1 and 4
Stage 4: Applied the newest vertex bisection three times(right) in congruence class 2 and 3.

$\triangle AXY, \triangle XBY, \triangle ZBY$ and $\triangle YZC$. Consider the Fig 8 (left). Since X, Y, Z be the midpoints of the edge AB, AC and BC, we have

$$XY \parallel BC, \quad ZY \parallel AB, \quad AX = BX, \quad BZ = CZ, \quad AY = CY$$

Since $XY \parallel BC$, we have $\angle AXY = \angle ABC$ and $\angle XYB = \angle ZBY$. Similarly, since $ZY \parallel AB$, we have $\angle YZC = \angle ABC$ and $\angle XBY = \angle ZYB$. Thus

$$\angle XYB = \angle ZBY, \quad |BY| = |BY|, \quad \angle XBY = \angle ZYB.$$

Therefore, we have $\triangle XBY \cong \triangle ZYB$, and we mark them in the congruency class 4. This further gives us $|AX| = |BX| = |YZ|$, and $|ZC| = |BZ| = |XY|$.

$$|AX| = |YZ|, \quad \angle AXY = \angle ABC = YZC, \quad |XY| = |ZC|.$$

Therefore, we have $\triangle AXY \cong \triangle YZC$. It's clear that $\triangle AXY$ and $\triangle YZC$ are similar to $\triangle ABC$ as all their angles are the same. Thus, we finished the proof of Claim 1.

Claim 2. Triangles with same number in Stage 4 in a same congruency class marked by the number.

Proof of Claim 2:
Let A, B, C be the vertices of the triangle T and let X, Y, Z be the midpoints of the edge AB, AC and BC, and M, N, P be the midpoints of the edge AY, CY and BY. Consider the Fig 8 (right).

An application of the newest vertex bisection refinement produces the following triangles

$$\triangle AXM, \triangle XBP, \triangle ZYP, \triangle YZN, \triangle MXY, \triangle PBZ, \triangle PYX, \triangle NZC$$

Notice that X, P and Z are three points one a stright line. This is clear because $XP \parallel AY$ and $PZ \parallel YC$, and we have $\angle BPX + \angle BPZ = \angle BYA + \angle BYC = \pi$. Basically, to prove the Stage 4 is equivalent to prove the following

$$\triangle AXM \cong \triangle XBP \cong \triangle ZYP \cong \triangle YZN$$
$$\triangle MXY \cong \triangle PBZ \cong \triangle PYX \cong \triangle NZC$$

Similarily to proof of Claim 1, we have

$$XY \parallel BC, \quad YZ \parallel AB, \quad XZ \parallel AC, \quad XM \parallel BY \parallel ZN$$

Therefore, we further have

$$\angle BAY = \angle ZYC, \quad \angle BCY = \angle XYA, \quad \angle AXC = \angle ABC = \angle YZC$$
$$\angle PZY = \angle ZYN, \quad \angle PYZ = \angle NZP, \quad \angle PXY = \angle XYM, \quad \angle PYX = \angle MXY,$$

Then it is clear that

$$\triangle AXY \sim \triangle ABC \sim \triangle YZC$$

Similarly, we can find that

$$\triangle XBZ \sim \triangle ABC, \quad \triangle AXM \sim \triangle ABY, \quad \triangle CZN \sim \triangle CBY.$$

In other words,

$$\triangle AXM \sim \triangle ABY \sim \triangle XBP \sim \triangle YZN,$$
$$\triangle CZN \sim \triangle CBY \sim \triangle YXM \sim \triangle ZBP.$$

Moreover, since the ratio of $\|AX\|, \|BX\|$, and $\|ZY\|$ is 1, and the ratio of $\|ZC\|, \|BZ\|$, and $\|XY\|$ is 1, we have

$$\triangle AXM \cong \triangle XBP \cong \triangle YZN,$$
$$\triangle CZN \cong \triangle YXM \cong \triangle ZBP.$$

Therefore, we showed that $\triangle AXM$, $\triangle XBP$ and $\triangle YZN$ are in congruency class 2, and $\triangle CZN$, $\triangle YXM$ and $\triangle ZBP$ are in congruency class 3. Moreover, since

$$\angle PZY = \angle ZYN, \quad \angle PYZ = \angle NZP, \quad \angle PXY = \angle XYM, \quad \angle PYX = \angle MXY,$$

we have that

$$\triangle YZN \cong \triangle YZP, \quad \triangle YXM \cong \triangle YXP.$$

Therefore we proved

$$\triangle AXM \cong \triangle XBP \cong \triangle ZYP \cong \triangle YZN \sim \triangle ABY,$$
$$\triangle MXY \cong \triangle PBZ \cong \triangle PYX \cong \triangle NZC \sim \triangle YBC.$$

Thus, we finished proof of Claim 2.

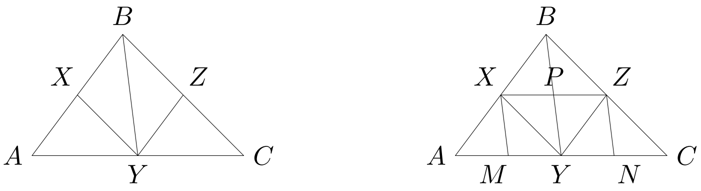

Figure 8: Illustration of newest vertex bisection for Claim 1(left); Claim 2(right).

Notice that in Stage 3, we see triangles in congruency class 1 again, so we can tell further applying the newest vertex bisection refinement will lead to the same process as what we have for Stage 1. Similarly, further applying the newest vertex bisection refinement over triangles in congruency class 4, 2 and 3 are already explored in Stage 3 and 4. Therefore, we actually obtain 4 congruency classes only. □

This means that we never have triangles degenerating when applying the newest vertex bisection refinement, because the number of congruence classes is four, which is finite, and by theorem proved in 3.1, we see that the newest vertex bisection refinement strategy is stable.

As we explain in section 3, a good refinement strategy should preserve both stability and consistency. Before we take a look at consistency, let's first introduce dependency graph.

5.1 Compatible Divisibility and Consistency

Definition 11. Let G = (N, A) be a simple directed graph, where N(nodes) represents triangles, and $(n_1, n_2) \in A(arrows)$ if the refinement edge of n_1 neighbors at n_2, and then we call G a dependency graph.

$$\begin{aligned} N &= \{T \in \mathcal{T} \;\mid\; \operatorname{diam} T = 2\}, \\ A &= \{(V_1, V_2) \in V \times V | \text{The refinement edge of } V_1 \text{ neighbors } V_2\}. \end{aligned}$$

Note that G is a simple directed graph, so it does not contain any loops. That is, for all $(v_1, v_2) \in A$, we have $v_1 \neq v_2$. Moreover, all nodes of one dependency graph have at most one outgoing arrow, because every triangle T has at most exactly one refinement edge(See 9). If the refinement edge of T borders no other triangles, then there is no arrow going from T to any other triangles in the dependency graph. If oppositely, it borders another triangle T' along its refinement edge, then there is an arrow in G going from T to T'.

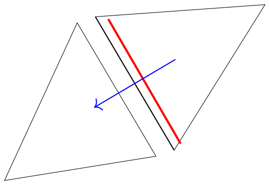

Figure 9: Illustration of outgoing arrow in dependency graph.

Lemma 8. A node n in the dependency graph has no outgoing arrow if and only if the corresponding triangle of n has a refinement edge at the boundary.

The proof for this is trivial based on the explanation for G as a simple graph.

Definition 12. A triangle is called compatibly divisible if either

a. it has no outgoing edge in G

b. it is part of a cycle in G whose size is 2.

Triangles of case b in Figure 10 are called mates as they share same refinement edge in 2-cycle.

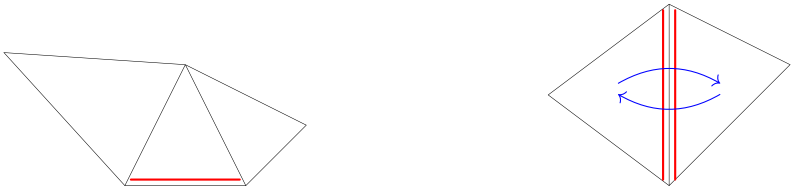

Figure 10: Compatibly divisible triangles: case a(left); case b(right)

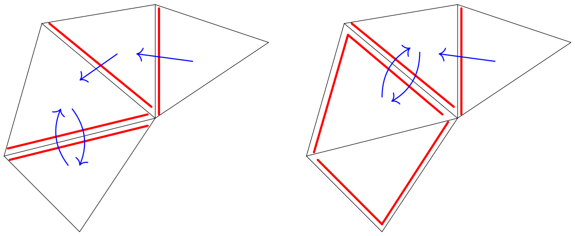

Figure 11: Compatible divisibility in dependency graph.

Compatible divisibility in Dependency Graph

In Figure 11, we see a compatibility chain in a triangulation. When performing the newest vertex bisection on the rightmost triangle, we need to bisect its left neighboring triangle first. We obtain a recursion here since we need to bisect the triangle which our current target triangle depends on. If we successfully reached the base case, either on the boundary or a cycle of size 2, we can then bisect back in an order like the stack. In the example displayed in Figure 11, a base case is reached by bisecting the leftmost two triangles. However, a base case is not always promised. In other words, it is not guaranteed that we can always achieve either bisection on the boundary or a cycle of size 2. One example is displayed in Figure 12, and this failed in applying the newest vertex bisection, because smaller triangles are dependent on each other. That is, its dependency graph is a cycle instead of a forest, i.e. collection of trees.

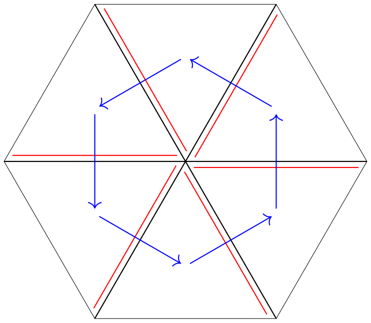

Figure 12: Failure of applying newest vertex bisection

Lemma 9. If the newest vertex bisection is only performed on

a. triangles isolated in the dependency graph

b. pairs of mates

then the refinement is stable and the triangulations will be consistent.

Proof. We already proved stability in general in two dimensions, and this applies here as well under a more strict condition. The consistency is obvious in compatibly divisible triangles. More specifically, if all triangles T of the initial simplicial complex $\mathcal{T}$ are compatibly divisible, then the number of recursions is bounded. That is, we never have a cyclic dependency graph like Figure 12. A detailed proof can be found in [10, 11]. □

Initial Refinement Edge
As described in the newest vertex bisection algorithm, the starting point is to pick the first peak which decides the first refinement edge. However, a decisive question in this process is how we choose the first peak and initial refinement edge to produce stable and consistent triangulation after each recursion. More specifically, since we know the newest vertex bisection preserves stability generally and consistency if its dependency graph has no cycles of length larger than two, the question is how to determine the initial refinement edge to promise a forest-like dependency graph.

One possibility is the following. We number the edges in any arbitrary manner. For each triangle, we always pick the edge with the highest number as refinement edge. Then there can not be a cycle of length greater than two in this dependency graph. One approach to have such a choice is to number the edges from shortest to longest. Then the longest edge of each triangle in the initial simplicial complex $\mathcal{T}$ as the refinement edge. The consistency is proved using this approach in Kossaczky's paper [8]. Basically, he proved that the dependency graph of $\mathcal{T}$ is acyclic with longest edge of triangles $T \in \mathcal{T}$ as refinement edge.

We may also ask whether it is even possible to choose initial refinement edges such that all triangles in the initial triangulation are compatibly divisible. In this case, the compatibility chain in the initial triangulation will have length zero (The more disconnected the dependency graph is, the better strategy is). Finding such initial choice of refinement edges is possible. The problem can be reduced to finding a perfect matching in a graph. A detailed work and proofs can be found in [10, 11, 12].

6 Discussion and Outlook

In this section, we take a brief look at difficulties in applications of uniform refinement strategy and newest vertex bisection in three dimensions. Changing focus from two dimensions to three or arbitrary higher dimension for mesh refinement, we are confronted with more challenges to preserve stability and consistency.

6.1 Uniform Refinement in 3D

While an application of uniform refinement on a simplicial complex in two dimensions is obviously stable and consistent, its application in three dimensions becomes more complicated. One observation is that the number of consistency classes after refinement of a single tetrahedron.

Let $T = [x^0, x^1, x^2, x^3]$ be the tetrahedron to be refined, and denote x^{ij} by the midpoint of the edge between x^i and x^j, for $0 \neq i < j \neq 3$. Uniform refinement of T produces eight different

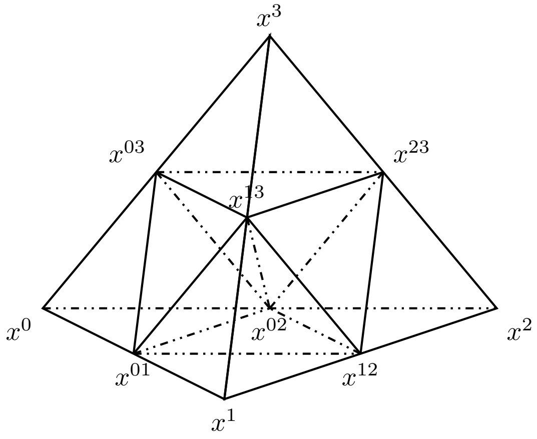

Figure 13: Uniform refinement in 3D

tetrahedron, which are given by

$$
\begin{aligned}
T_1 &:= [x^0, x^{01}, x^{02}, x^{03}], & T_2 &:= [x^{01}, x^1, x^{12}, x^{13}],\\
T_3 &:= [x^{01}, x^{02}, x^{03}, x^{13}], & T_4 &:= [x^{01}, x^{02}, x^{12}, x^{13}],\\
T_5 &:= [x^{02}, x^{12}, x^2, x^{23}], & T_6 &:= [x^{02}, x^{12}, x^{13}, x^{23}],\\
T_7 &:= [x^{02}, x^{03}, x^{13}, x^{23}], & T_8 &:= [x^{03}, x^{13}, x^{23}, x^3].
\end{aligned}
$$

With similar proofs, we can see that these eight tetrahedron belong to at most two congruence classes. We have

$$
\begin{aligned}
T_1 \cong T_2 \cong T_5 \cong T_8,\\
T_3 \cong T_4 \cong T_6 \cong T_7.
\end{aligned}
$$

In summary, we can still easily obtain consistency in this global refinement, but checking stability becomes more complex. By the theorem in section 3.2, we know that if the number of congruence classes is finite, then the refinement strategy is stable. This implies that one difficulty of understanding the uniform refinement in arbitrary higher dimension is counting and checking congruency classes.

6.2 Newest Vertex Bisection in 3D

The newest vertex bisection becomes more complicated in its application in three dimension. One method is discussed in an article by Arnold, Mukherjee and Pouly [1]. Basically, a tetrahedron T is classified into four types: planar, adjacent, opposite and mixed, based on their refined edges and faces. There are two steps in his algorithm. The first step is to bisect every tetrahedron marked for refinement on their classification; and second is to preserve consistency with additional bisection to ensure all tetrahedra are consistent without hanging nodes.

6.3 Longest Edge Bisection

Besides newest vertex bisection strategy, there exist other bisection methods for mesh refinement. One method is the longest edge bisection proposed by Rivara [14]. While the refinement edge in the newest vertex bisection is chosen opposite to the newest vertex, it is now determined by the length of each edge in longest edge bisection: always bisecting the longest edge in each triangle. See Figure 14.

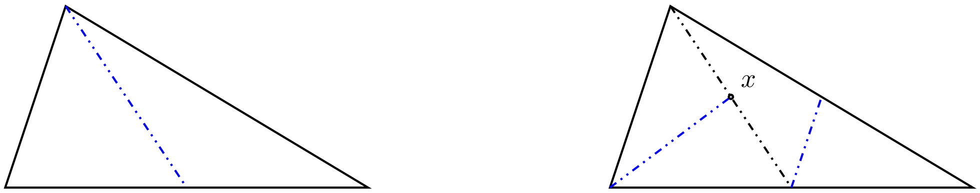

Figure 14: Longest edge bisection in 2D

In two dimensions, the longest edge bisection is stable. As proven by Stynes, there appear only a finite number of congruency classes [15, 16, 17]. We notice that unfortunately, the consistency requires additional bisections to be made, similar to newest vertex bisection. For example, x in Figure 14 is a hanging node.

References

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

[2] Randolph E Bank and L Ridgway Scott. On the conditioning of finite element equations with highly refined meshes. *SIAM Journal on Numerical Analysis*, 26(6):1383–1394, 1989.

[3] Randolph E Bank, Andrew H Sherman, and Alan Weiser. Some refinement algorithms and data structures for regular local mesh refinement. *Scientific Computing, Applications of Mathematics and Computing to the Physical Sciences*, 1:3–17, 1983.

[4] J. Bey. Tetrahedral grid refinement. *Computing*, 55(4):355–378, Dec 1995.

[5] Jürgen Bey. Simplicial grid refinement: on freudenthal's algorithm and the optimal number of congruence classes. *Numerische Mathematik*, 85(1):1–29, 2000.

[6] Philippe G Ciarlet. The finite element method for elliptic problems. *Classics in applied mathematics*, 40:1–511, 2002.

[7] Roberto Grosso and Günther Greiner. Hierarchical meshes for volume data. In *Computer Graphics International, 1998. Proceedings*, pages 761–769. IEEE, 1998.

[8] Igor Kossaczkỳ. A recursive approach to local mesh refinement in two and three dimensions. *Journal of Computational and Applied Mathematics*, 55(3):275–288, 1994.

[9] Michal K˘rí˘zek and Theofanis Strouboulis. How to generate local refinements of unstructured tetrahedral meshes satisfying a regularity ball condition. *Numerical Methods for Partial Differential Equations*, 13(2):201, 1997.

[10] William F Mitchell. *Unified multilevel adaptive finite element methods for elliptic problems*. PhD thesis, University of Illinois at Urbana-Champaign Urbana, IL, 1988.

[11] William F Mitchell. Adaptive refinement for arbitrary finite-element spaces with hierarchical bases. *Journal of computational and applied mathematics*, 36(1):65–78, 1991.

[12] William F Mitchell. 30 years of newest vertex bisection. In *AIP Conference Proceedings*, volume 1738, page 020011. AIP Publishing, 2016.

[13] Douglas W Moore. Simplical mesh generation with applications. Technical report, Cornell University, 1992.

[14] Maria-Cecilia Rivara. Mesh refinement processes based on the generalized bisection of simplices. *SIAM Journal on Numerical Analysis*, 21(3):604–613, 1984.

[15] Martin Stynes. An n-dimensional bisection method for solving systems of n equations in n unknowns. *Applicable Analysis*, 9(4):295–296, 1979.

[16] Martin Stynes. On faster convergence of the bisection method for certain triangles. *Mathematics of Computation*, 33(146):717–721, 1979.

[17] Martin Stynes. On faster convergence of the bisection method for all triangles. *Mathematics of Computation*, 35(152):1195–1201, 1980.

[18] Shangyou Zhang et al. Successive subdivisions of tetrahedra and multigrid methods on tetrahedral meshes. *Houston J. Math*, 21(3):541–556, 1995.