

# AN ALGORITHM FOR TWO-DIMENSIONAL MESH GENERATION BASED ON THE PINWHEEL TILING\*

Pritam Ganguly<sup>1</sup>

Stephen A. Vavasis<sup>2</sup>

Katerina D. Papoulia<sup>3</sup>

<sup>1</sup>*Department of Theoretical and Applied Mechanics, Thurston Hall, Cornell University, Ithaca, NY 14853, U.S.A., pg45@cornell.edu.*

<sup>2</sup>*Department of Computer Science, Upson Hall Cornell University, Ithaca, NY 14853, U.S.A., vavasis@cs.cornell.edu.*

<sup>3</sup>*School of Civil and Environmental Engineering, Hollister Hall, Cornell University, Ithaca, NY 14853, U.S.A., kp58@cornell.edu.*

## ABSTRACT

We propose a new two-dimensional meshing algorithm called PINW able to generate meshes that accurately approximate the distance between any two domain points by paths composed only of cell edges. This technique is based on an extension of pinwheel tilings proposed by Radin and Conway. We prove that the algorithm produces triangles of bounded aspect ratio. This kind of mesh would be useful in cohesive interface finite element modeling when the crack propagation path is an outcome of a simulation process.

**Keywords:** Pinwheel tiling, mesh generation, isoperimetric, cohesive interface finite element

## 1. INTRODUCTION

One of the most widely used techniques to simulate fracture is cohesive interface finite element modeling. In this kind of model, the area or volume under consideration is subdivided into bulk elements, which are typically triangles or quadrilaterals in 2D and tetrahedra or hexahedra in 3D. Next, interfacial elements, which are edge elements in 2D or surface elements in 3D, are placed between some or all pairs of adjacent bulk elements. The cohesive model prescribes a relationship relating traction to relative displacement on the interfacial elements. There is an abundance of literature that deals with the nature of this relationship, e.g., see [1] and the references therein. A widely accepted modeling assumption is that the total energy to create the crack is proportional to its surface area (or length in 2D). In fact, the critical energy release rate  $G_c$  per unit surface area or length of crack is often

a parameter of the cohesive model.

In a finite element model, the energy release rate is associated with surface area or length of interfacial elements composing the crack being modeled. If the discrepancy between the “true” crack path (i.e., the path the crack would follow if it were not for the finite element constraint that the crack path must lie on predetermined interfacial elements) and the path of the simulated crack is large for certain paths, then nonphysical preferred crack directions can exist. In other words, the results of the simulation would depend upon how well the boundaries of the mesh cells are aligned along the true crack path. In this paper, we propose a meshing technique that approximates the true path with the path along mesh boundaries with high accuracy even though the true path is unknown to the mesh generation algorithm. In particular, the approximation has the property that for any crack path, the simulated and true crack path lengths converge to each other upon refining the mesh, which is a property not possessed by other simpler families of meshes. We

---

\*Supported in part by NSF Grants CMS-0239068 and CCF-0085969

call this algorithm the PINW mesh generator because it is based on an extension of the 1:2 pinwheel tiling described in the next section.

In Section 3 we define “deviation ratio” and consider a simple experiment to test the properties of the 1:2 pinwheel mesh. The 1:2 pinwheel tiling seems to be too restricted to be useful for a general-purpose algorithm, so we explain how to generalize it to arbitrary triangles in Section 4. This generalization is the basis for our meshing algorithm PINW. In Section 5 we prove that our generalization still has the isoperimetric property. Then in Section 6 we describe the algorithm. The main new ingredient introduced in that section is a procedure to convert a tiling to a mesh. The aspect ratio of the resulting mesh is analyzed in Section 7.

The aspect ratio of the mesh is important for the cohesive fracture application because the bulk elements (e.g., triangles in 2D) are used to model a continuum mechanical theory such as linear elasticity. It is well-known (see, e.g., Theorem 4.4.4 of [2], in which aspect ratio is called “chunkiness”) that poorly shaped elements can lead to substantial errors in the elasticity solution.

## 2. PINWHEEL TILINGS

In this section, we provide a brief introduction to the properties of pinwheel tilings. Tilings are a covering of the euclidean 2-space  $\mathcal{E}^2$  starting with a finite number of shapes called *prototiles*. The tilings are constructed by translated and rotated copies of the prototiles that intersect each other only along the boundaries. The tilings were proposed to model crystallographic structures in the physics community.

The pinwheel tilings [3] are classified as *aperiodic tilings*. In  $\mathcal{E}^2$  this is equivalent to saying that no translation of the tiling leaves it invariant. The basic pinwheel tiling as developed by Radin and Conway has a hierarchical structure and is constructed by successive operations of subdivisions and expansions.

Consider a right triangle with legs of length 1 and 2 referred to as the *short* and *medium* sides. The hypotenuse is thus of length  $\sqrt{5}$  and will be called the *long* edge. The vertices will be named similarly, that is, the *small*, *medium* and *long* vertices are opposite the corresponding sides. For brevity, we will call a right triangle with the ratio of its short to medium edge equal to 1/2 as a “1 : 2 right triangle” and the tiling formed by its copies as a “1 : 2 tiling.” This single tile is subdivided into five triangles that are all congruent to each other as shown in Figure 1.

If one were to dilate the subdivision in Figure 1 by a factor of  $\sqrt{5}$  and then rotate and translate the resulting figure so that the dilated copy of *C* ended up coin-

cident with the original tile *P*, then a larger subset of  $\mathcal{E}^2$  would now be tiled. The above subdivision scheme is then applied to each of the five copies of *P*, and then another dilation followed by rotation and translation is carried out. Continuing this process infinitely would lead to a tiling of the plane. Thus, in the case of the standard pinwheel tilings, *P* and *P<sub>R</sub>* (where *P<sub>R</sub>* denotes the reflection of *P* about the x-axis) form the set of fixed prototiles and the tiling uses translations and rotations of this set.

For our purposes however, we will concentrate just on the subdivision step and omit the dilation, translation and rotation steps, leading to the “subdivision” pinwheel tiling in which the cell diameter tends to zero and the area of the plane covered by the mesh does not expand from step to step. This is because we are interested in generating a mesh with varying amounts of refinement for a fixed region rather than a mesh that ultimately covers  $\mathcal{E}^2$ . In the subdivision pinwheel tiling, one starts with a fixed 1 : 2 triangle and then repeatedly subdivides first the initial triangle and then each subtriangle into five congruent subtriangles using the above rule.

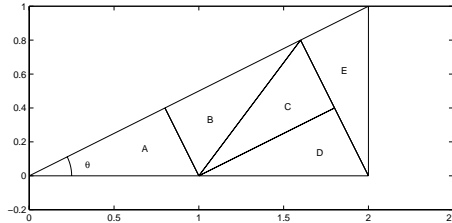


Figure 1: Basic pinwheel subdivision proposed by Radin.

One can enumerate the rotation angles of the child triangles with respect to *P* and *P<sub>R</sub>* as  $R_\theta P_R$ ,  $R_{\pi+\theta} P$ ,  $R_\theta P$ ,  $R_\theta P_R$ ,  $R_{\frac{\pi}{2}+\theta} P$  where  $R_\theta$  is rotation by  $\theta$  in the counterclockwise direction. For the standard 1 : 2 right triangle,  $\theta = \arctan(1/2)$  and  $\theta/\pi$  in this case is irrational. The significance of this is as follows. As the number of subdivisions goes to infinity, so do the distinct orientations of the triangles. For example, suppose we keep track of the orientation of all triangles of type *C* with respect to the parent triangle in the subdivisions. As can be seen in Figure 1, the angle made by a triangle of type *C* with respect to the parent in the *n*th subdivision is  $n\theta$ . Since  $\theta/\pi$  is irrational,  $n\theta$  will represent a different angle for each *n*.

This presence of an infinite number of orientations leads to a special property known as the isoperimetric property [4]. For a tiling of  $\mathcal{E}^2$ , *isoperimetry* means that given an  $\epsilon > 0$ , there exists an *R* such that for any two points *P* and *Q* on the boundaries of the triangles with  $\|P - Q\| > R$ , the shortest path from *P* to *Q* that

uses only tile edges has length at most  $(1 + \epsilon)\|P - Q\|$ . Here  $\|P - Q\|$  denotes the Euclidean distance from  $P$  to  $Q$ , which will also be denoted as  $|PQ|$ .

There is an analogous property for the subdivision pinwheel tiling. In this case, let  $P, Q$  be two points on the boundary of the initial triangle. Then for every  $\epsilon > 0$ , there exists an  $n$  such that after  $n$  recursive subdivisions of the initial triangle, the shortest path from  $P$  to  $Q$  using only triangle edges is at most  $\|P - Q\|(1 + \epsilon)$ . This theorem can be generalized so that  $P$  and  $Q$  do not have to be on the boundary of the initial triangle but may be any two distinct points.

The isoperimetric property is the reason that pinwheel tilings are attractive for cohesive interface modeling. Consider a finite region  $\Omega \subset \mathcal{E}^2$  tiled with an infinite sequence of pinwheel tilings  $\mathcal{M}_1, \mathcal{M}_2, \dots$  in which the triangles in  $\mathcal{M}_i$  all have side lengths  $h_i, 2h_i, \sqrt{5}h_i$ , and  $h_i \rightarrow 0$  as  $i \rightarrow \infty$ . Then for an arbitrary straight segment of length  $l$  connecting  $p \in \Omega$  to  $q \in \Omega$ , and for an arbitrary  $\epsilon > 0$ , there exists an  $I$  such that in each of the tilings  $\mathcal{M}_I, \mathcal{M}_{I+1}, \dots$ , there is a path from  $p$  to  $q$  using only mesh edges (except for initial and ending segments to connect  $p$  and  $q$  to the boundaries of the triangles that contain them) such that the length of the path is  $l(1 + \epsilon)$ . We will state this result in a more general setting in Section 5.

Since the above result holds for an arbitrary line segment, it also holds for any piecewise smooth curve or network of such curves. The reason is that a network of piecewise smooth curves can be approximated arbitrarily accurately with a network of line segments. Then each of the line segments can be approximated arbitrarily accurately with paths of the pinwheel tiling.

Thus, when used for cohesive fracture, the pinwheel tiling has the property that all possible crack paths are approximated as accurately as desired (in terms of their length) by paths that use only mesh edges, as the mesh diameter tends to zero. As we shall see in the next section, more common mesh generation techniques do not have this property.

### 3. A COMPUTATIONAL EXPERIMENT

In this section we carry out some simple experiments to quantify the isoperimetric property of the 1:2 tiling. Since our interest here is in meshes, we first explain how to convert the 1:2 pinwheel tiling to a mesh. It is apparent from Figure 1 that the pinwheel tiling is almost a triangulation except for the hanging node bisecting the medium side of triangle  $E$ . We define a *hanging node* of a planar subdivision into triangles to be a point that is a vertex of one triangle but lies on the strict relative interior of an edge of another triangle.

**Table 1:** Direct computation of deviation ratios for the first five levels of pinwheel subdivision.

$n$	$\text{dev}_1(\mathcal{PT}_n)$
1	1.3416
2	1.1948
3	1.1843
4	1.1264
5	1.0831

It is fairly simple to make the pinwheel tiling a mesh [3]: we divide every triangle into two by joining its medium vertex to the midpoint of its medium edge. In fact, it is not necessary to split all the triangles, and in our example we have obtained a mesh by splitting a certain subset of the tiles. This splitting is done only on the finest level of the pinwheel subdivision.

Our computational experiment is as follows. Starting from a 1 : 2 rectangle, we divide it into two 1 : 2 triangles and then apply the pinwheel subdivision  $n$  times to each of the 1 : 2 triangles. Thus, the final tiling has  $2 \cdot 5^n$  triangles. The resulting tiling of the original rectangle is then converted to a mesh using the technique in the last paragraph.

Given a tiling  $\mathcal{T}$  of a domain  $\Omega$ , let  $\text{Skel}(\mathcal{T})$  be the 1-skeleton of  $\mathcal{T}$ , that is, the union of all edges of all triangles, and let  $V(\mathcal{T})$  be the set of all vertices of  $\mathcal{T}$ . Let  $l$  be a positive parameter chosen small enough so that  $\Omega$  contains a disk of diameter  $l$ . We propose to evaluate isoperimetric quality of the triangulation with the following quantity, which we refer to as the *l-path deviation ratio*:

$$\text{dev}_l(\mathcal{T}) = \max \left\{ \frac{\text{dist}_{\text{Skel}(\mathcal{T})}(p, q)}{\|p - q\|} : p, q \in V(\mathcal{T}) \text{ and } \|p - q\| \geq l \right\}$$

Here,  $\text{dist}_{\text{Skel}(\mathcal{T})}(\cdot, \cdot)$  means shortest distance among paths restricted to  $\text{Skel}(\mathcal{T})$ . Thus, this quantity measures the maximum ratio between the paths in the mesh versus Euclidean paths. Clearly for any mesh  $\mathcal{T}$  of any polygon,  $\text{dev}_l(\mathcal{T}) > 1$ . The pinwheel mesh of the 1:2 rectangle has the property that for any  $l \in (0, 1)$ ,  $\text{dev}_l(\mathcal{PT}_m) \rightarrow 1$  as  $m \rightarrow \infty$  where  $\mathcal{PT}_m$  is the pinwheel tiling of the 1 : 2 rectangle after  $m$  levels of refinement.

Our experiment is to evaluate  $\text{dev}_1(\mathcal{PT}_m)$  for  $\mathcal{PT}_1, \dots, \mathcal{PT}_5$ . The results are depicted in Table 1.

In contrast, consider the meshes in Figure 2. The deviation ratios of these meshes have lower bounds greater than 1 irrespective of the number of subdivisions. In particular, the lower bound is  $\sqrt{2} \approx 1.414$  for

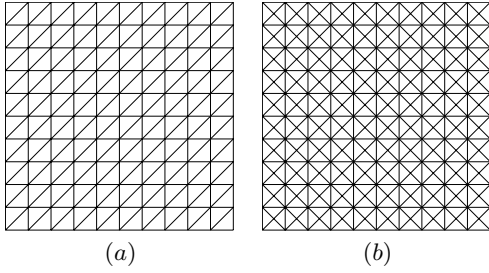


Figure 2: Two regular meshes of a square

the mesh in Figure 2(a). For the mesh that was used by Xu and Needleman [5] (one of the first papers on cohesive finite element modeling), which is shown in Figure 2(b) and is sometimes called a “cross-triangle quadrilateral” mesh, the worst case deviation ratio can be shown to be approximately equal to 1.082 in the limit as the mesh cell size tends to 0.

#### 4. GENERALIZATION OF PINWHEEL TILINGS

The 1 : 2 pinwheel tiling discussed up to now was extended to a tiling with an arbitrary right triangle and its reflection as a prototiles by Sadun [6]. The small angle of the prototile determines the finiteness of the orientations and sizes of the tiles in the tilings that are discussed in [6]. We now describe our approach to extend the pinwheel subdivision to arbitrary (non-right) triangles.

First we propose a way of subdividing a general triangle and show that any number of subdivisions would produce triangles similar to a finite set of prototiles. Consider the triangle shown in Fig. 3. We denote the vertices by  $A$ ,  $B$  and  $C$  in clockwise order and the included angles at these vertices by  $a$ ,  $b$  and  $c$  respectively. Assume also  $a < c$ . First, draw the segment  $CF$  such that  $F$  is a point on  $AB$  and  $\angle FCB = a$  measured counterclockwise from  $CB$ . From  $F$  draw  $FD$  such that  $D$  is on  $AC$  and  $\angle DFC = b$  measured clockwise from  $FC$ . From  $D$  draw  $E$  and  $G$  such that  $E$  is on  $AB$  and  $G$  is on  $CF$  and  $\angle ADE = b$  clockwise from  $DA$  and  $\angle GDC = a$  counterclockwise from  $DC$ . Thus, we have a subdivision of a general  $\triangle ABC$  into five triangles of which  $I$ ,  $III$  and  $V$  are similar to the parent and the remaining two  $II$  and  $IV$  are similar to each other but not to the parent. Note that we required  $a < c$  to make this construction but we did not require any ordering on  $b$ .

**Theorem 1.** *The above procedure for subdivision produces triangles with angles belonging to the set  $A_1 = \{a, b, c\}$  or to the set  $A_2 = \{a, c - a, \pi - c\}$ .*

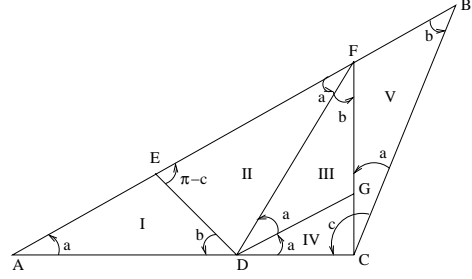


Figure 3: Our generalized pinwheel subdivision of an arbitrary triangle into five subtriangles.

*Proof.* This is obvious by simply checking all the angles in Figure 3 and using the fact that angles of a triangle sum to  $\pi$ .  $\square$

**Theorem 2.** *If the above subdivision procedure is used recursively on the subtriangles, then any triangle produced has angles either from  $A_1$  or  $A_2$ .*

*Proof.* One checks that if we define  $a' = a$ ,  $b' = c - a$ ,  $c' = \pi - c$  then  $\{a', c' - a', \pi - c'\} = \{a, b, c\}$ .  $\square$

For the rest of this paper, we say that a triangle with angles  $\{a, b, c\}$  (listed in this order) is *conjugate* to a triangle with angles  $\{a, c - a, \pi - c\}$ . The point of Theorem 2 is that conjugacy is a symmetric relationship. We remark that if the original triangle is a right triangle, i.e.,  $c = \pi/2$ , then this triangle is similar to its conjugate. This is the case considered by [6].

These two theorems imply a procedure for subdividing any initial triangle  $T_1 = \triangle ABC$  with angles  $a, b, c$ . Assume  $a \leq b \leq c$ . Apply the first subdivision rule to get five smaller triangles. Then, for the three similar to  $T_1$ , reapply the same rule recursively. For the two conjugates, apply the other rule. For the conjugate triangles, we do not necessarily have the order  $a \leq c - a \leq \pi - c$ , but we do not need that order. We need only the inequality  $a < \pi - c$ , which must be true since  $a + b + c = \pi$ .

This procedure runs into a difficulty when  $c \approx a$  (i.e., the initial triangle is close to equilateral) because in this case the conjugate triangle will have a bad aspect ratio. We get around this problem as follows. If  $c \approx a$ , then we first subdivide the initial triangle into three about its in-center, that is, we join the in-center to the vertices of the original triangle and form three subtriangles. We use a cutoff in our algorithm: if  $c - a$  is less than the cutoff, then the preliminary tripartition is carried out. The cutoff for  $c - a$  is chosen to optimize the smallest angle. In other words, a parent is divided about the in-center if the smallest angle prior to division is smaller than after the division. Here smallest

angle happens to be the minimum of the angles in the two sets  $A_1$  and  $A_2$  for a given set of angles  $\{a, b, c\}$  and can be shown to be  $\approx 0.4$  rad. Thus, we take the cutoff to be 0.4 rad.

## 5. ISOPERIMETRIC PROPERTY

This section is devoted to showing the result that the generalization of the pinwheel tiling introduced in the previous section obeys an isoperimetric inequality. The analysis and proof technique in this section closely follow the proof from [4]. The following is the key lemma in the proof of isoperimetry.

**Lemma 1.** *Let triangle  $T = \triangle ABC$  be as above. Assume  $a/\pi$  is an irrational number, where  $a$  is the angle of  $T$  at  $A$ . Let  $\theta \in [0, 2\pi)$  and  $\epsilon > 0$  be arbitrary. Then there is a refinement of  $T$  following the above rules that contains a triangle edge  $e$  such that the angle between  $e$  and the  $x$ -axis lies in the interval  $(\theta - \epsilon, \theta + \epsilon)$ .*

*Furthermore, the length of  $e$  is at least  $\zeta(a, b, c, \epsilon)L$ , where  $a, b, c$  are the angles of  $T$ ,  $\epsilon$  is as above,  $\zeta(\cdot)$  is a fixed positive-valued function, and  $L$  is the longest side-length of  $T$ .*

*Proof.* Observe that Triangle III in the above subdivision is similar to the initial triangle  $T$  but is rotated by angle  $a$ . Call this triangle  $T'$ . If this triangle is subdivided by the same rule again, there will be another smaller copy of  $T$ , say  $T''$ , rotated by  $2a$  etc. The infinite sequence  $a, 2a, 3a, \dots$  taken mod  $2\pi$  is dense in the interval  $[0, 2\pi)$  by the assumption that  $a/(2\pi)$  is irrational. Therefore, for some sufficiently fine mesh, there is an edge  $e$  of triangle  $T^{(k)}$  in the interval  $(\theta - \epsilon, \theta + \epsilon)$ .

For the second part of the lemma, observe that for any  $\epsilon > 0$  there is an  $n \equiv n(\epsilon, a)$  such that every point in  $[0, 2\pi]$  is distance (mod  $2\pi$ ) at most  $\epsilon$  from at least one point in the set  $\{a, 2a, 3a, \dots, na\}$ . Therefore, one of  $T, T', \dots, T^{(n)}$  described in the last paragraph will have the desired edge  $e$ . The longest side-length of  $T$  is  $L$ ; the longest side-length of  $T'$  is  $q(a, b, c)L$ , where  $q$  is some universal function (not depending on anything other than  $a, b, c$ ) derived from our construction. By similarity, the longest edge of  $T^{(2)}$  has length  $q(a, b, c)^2L$ . Thus, if we define  $\zeta(a, b, c, \epsilon) \equiv \gamma(a, b, c)q(a, b, c)^{n(\epsilon, a)}$ , where  $\gamma(a, b, c)$  is the ratio of the shortest to longest side length of  $T$ , then the length of  $e$  is at least  $\zeta(a, b, c, \epsilon)L$ . This proves the second part of the lemma is also satisfied.  $\square$

For the first main theorem of this section, we need one more definition. We say that a generalized tiling  $T'$  of a triangle  $T$  refines another generalized tiling  $T$  of  $T$  provided that for each tile  $\tau$  of  $T$ , either  $\tau$  appears in  $T'$  or a subdivision of  $\tau$  appears in  $T'$ . This definition

implies that  $V(T) \subset V(T')$  and  $\text{Skel}(T) \subset \text{Skel}(T')$ . The first main theorem for this section is as follows.

**Theorem 3.** *Let  $T = \triangle ABC$  be a triangle with angles  $a, b, c$  such that  $a < c$  and  $a/\pi$  is irrational. Let  $T^0, T^1, \dots$  be an infinite sequence of generalized tilings of  $T$  generated by the rules above. For each  $i$ , let  $y_i$  be the maximum tile diameter in  $T^i$ . We assume the sequence of tilings has the following two properties: (a)  $T^{i+1}$  refines  $T^i$ , and (b)  $y_i \rightarrow 0$  as  $i \rightarrow \infty$ . Let  $P, Q$  be any two points on the boundary of  $T$ . Then*

$$\lim_{i \rightarrow \infty} \text{dist}_{\text{Skel}(T^i)}(P, Q) = |PQ|.$$

In other words, every straight-line path connecting two points  $(P, Q)$  on the boundary of  $T$  is approximated with arbitrary accuracy by a path of edges of the tiling.

Due to space limitations, we defer the proof of this theorem to the full paper. A draft of the full paper is available from arxiv.org.

The preceding theorem has the drawback that it pertains only to paths starting and ending on the boundary of the root triangle. For isoperimetry, we would like to generalize the result to paths with arbitrary interior  $P$  and  $Q$ . Since the nodes of the pinwheel tiling are dense in the interior (in the limit as the mesh size is refined), the following theorem provides a suitable generalization and will be taken as our definition of the isoperimetric property.

**Theorem 4.** *Let  $T^0, T^1, \dots$  be a sequence of generalized pinwheel tilings of  $T$  (satisfying  $a < c$  and  $a/\pi$  is irrational as in the previous theorem) such that the maximum cell diameter tends to zero and such that  $T^{i+1}$  refines  $T^i$  for all  $i = 0, 1, 2, \dots$ . Let  $P, Q$  be any pair of distinct points lying on  $\text{Skel}(T^n)$  for some  $n$ . Then*

$$\lim_{\substack{m \rightarrow \infty \\ m \geq n}} \text{dist}_{\text{Skel}(T^m)}(P, Q) = \|P - Q\|.$$

*Proof.* Consider the segment  $PQ$  lying in  $T$ . Let  $\epsilon > 0$  be given. Make a list  $U_1, \dots, U_r$  of tiles in  $T_n$  traversed by this segment. Since  $PQ$  crosses  $U_i$ , define  $P_iQ_i$  to be  $U_i \cap PQ$ . Observe that  $P_i, Q_i$  both lie on the boundary of  $U_i$ . By the preceding theorem, after a sufficient number of further subdivisions (say  $s$ ), there exists a path in  $\text{Skel}(T^{n+s})$  between  $P_i$  and  $Q_i$  of length  $|P_iQ_i|(1 + \epsilon)$ . This choice of  $s$  depends on  $i$ , so take the maximum such value of  $s$  (maximum over all  $i = 1, \dots, r$ ). Then there is a path in  $\text{Skel}(T^{n+s})$  from  $P$  to  $Q$  of length at most

$$|P_1Q_1|(1 + \epsilon) + |P_2Q_2|(1 + \epsilon) + \dots + |P_rQ_r|(1 + \epsilon),$$

i.e., at most  $|PQ|(1 + \epsilon)$ .  $\square$

## 6. MESHING AN ARBITRARY REGION

In this section we present our algorithm PINW to mesh a region  $\Omega$  with arbitrary polygonal boundary. A summary of PINW appears in Figure 4. The steps in this summary are described in more detail in the remainder of this section. The current version of PINW is 1.0 and has been coded in Matlab. An example output from this algorithm is shown in Fig. 5.

We first start with a coarse triangulation of the domain. We use the *Triangle* package [7] developed by J. Shewchuk, which uses Delaunay triangulation. The triangles produced have bounded aspect ratio. The second preliminary step, as mentioned in Section 4, locates triangles too close to equilateral and splits them at their in-center.

A third preliminary step is to identify and split triangles whose smallest angle  $a$  is a rational multiple of  $\pi$ . In principle, this test could be conducted exactly using number-theoretic methods since the coordinates of the vertices of each triangle, being floating-point numbers, are rational numbers and can be treated with integer algorithms by clearing common denominators. Modifying a triangle in which  $a$  is a rational multiple of  $\pi$  is trivial in principle because any small random perturbation of a node of such a triangle will lead to an angle that is not a rational multiple of  $\pi$  with probability 1.

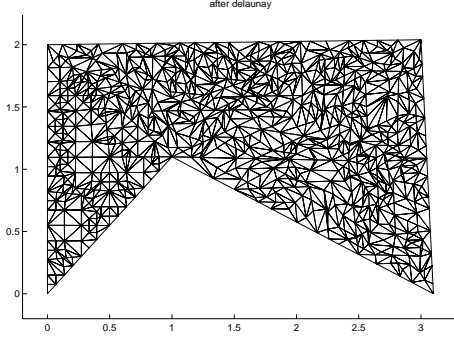
In practice, this exact test and solution are both undesirable. For practical use of the algorithm, we would like to avoid the case when  $a$  is close to a rational multiple of  $\pi$  of the form  $m\pi/n$  where  $n$  is a small integer. Therefore, a more practical heuristic is to check each smallest angle against a finite list of the form  $m\pi/n$ , where  $m, n$  range over a pre-selected set of small integers. This step has not been implemented in the current version of our code PINW 1.0 because we are still seeking the best practical heuristic. (Indeed, in Figure 5, one coarse triangle is close to a 45-degree right triangle, and hence one part of the subdivision exhibits a shortage of possible directions.)

Let  $\mathcal{T}_0$  be the list of triangles that are produced by these preliminary steps. Thus, the triangles in  $\mathcal{T}_0$  form a simplicial triangulation of the input set  $\Omega$ . We call these triangles the *root tiles*. The generalized pinwheel subdivision is then performed on the the root tiles to obtain a refined tiling. The procedure to refine the mesh used in PINW 1.0 is based on a simple heap [8]. The heap is initialized with all triangles in  $\mathcal{T}_0$ , which are ordered in the heap according to length of the minimum altitude. The main loop for the subdivision is to remove the top member of the heap (i.e., the unsubdivided tile with the largest value of minimum altitude) and replace it with its five children. The procedure terminates when the top triangle in the heap is

### Algorithm PINW 1.0

1. Generate a mesh for  $\Omega$  with bounded aspect ratio using Triangle.
2. Split triangles too close to equilateral at their in-centers.
3. Split triangles whose smallest angle is a rational multiple of  $\pi$  at a point near the in-center.
4. Let the set of triangles obtained after steps 1–3 be called  $\mathcal{T}_0$ .
5. Initialize a heap containing triangles that need splitting. The triangles are ordered so that the one whose minimum altitude is maximum is at the top of the heap. Initially the heap contains all triangles from  $\mathcal{T}_0$ .
6. Repeatedly remove a triangle from the heap and split it into five, until the size of the top element of the heap is sufficiently small according to the user’s specification.
7. Let  $\mathcal{T}_*$  be the set of tiles including those in  $\mathcal{T}_0$  and all their descendants obtained by subdivision. Let  $\mathcal{T}_f \subset \mathcal{T}_*$  be the set of leaf tiles.
8. Loop over all tiles in  $\mathcal{T}_*$  starting from the coarsest to determine the value of  $\text{big}(e)$  for each edge  $e$  of any tile.
9. For each big edge (i.e., each edge in the image of the “big” operator), select one side as *moving* and the other as *staying*. Sort the list of nodes lying on the staying side of each such edge.
10. Loop over tiles in  $\mathcal{T}_*$  starting from the coarsest excluding  $\mathcal{T}_f$ . For each such tile  $T$  and for each of its vertices  $D, E, F$  as labeled in Figure 3, let  $e$  be the maximal big edge containing the particular vertex. If this vertex  $D, E$  or  $F$  is on the moving side of  $e$  and is very close to a vertex  $v'$  on the staying side, then displace it to coincide with  $v'$  and apply the induced affine transformation to subtriangles of  $T$ .
11. Apply Delaunay triangulation to each distorted, subdivided leaf tile. (The distortion of the leaf tiles is due to the affine transformations in the previous step. The subdivision of the edges is due to the presence of hanging nodes.) The collection of triangles output from this step is a simplicial mesh of  $\Omega$ .

Figure 4: Overview of the steps of the PINW algorithm.



**Figure 5:** The coarse mesh for this figure had three triangles. The final mesh after pinwheel tiling, collapse-node operations and Delaunay triangulation is shown.

smaller than the user-specified mesh size requirement.

Note that during the subdivision procedure, the angles  $a, b, c$  in Figure 3 are assigned to smallest, middle and largest angles respectively for tiles similar to root tiles. For the conjugate tiles, angles  $a, b, c$  are assigned according to the conjugacy relationship. In other words, if the angles of the root tile are  $a', b', c'$  in that order, then the angles  $a, b, c$  in the conjugate tile are assigned in the order  $a = a', b = c' - a',$  and  $c = \pi - c'$ . This ensures that the conjugate of the conjugate is again similar to the root tile.

From this description, it is apparent that PINW 1.0 supports a single global user-specified mesh size requirement. For many applications of mesh generation, it is useful to have a finer mesh in one part of the domain versus another. This can also be implemented in the framework of generalized pinwheel subdivision but is not available in PINW 1.0. In addition, several aspects of our analysis that follows below would have to be generalized to cover graded meshes.

Once the subdivision procedure is complete, the resulting tiling must be converted to a simplicial mesh. For the 1:2 pinwheel triangulation, this step is quite straightforward as mentioned in Section 3. In the generalized case, however, it is much more complicated and involves several steps that we shall now describe.

Let  $\mathcal{T}_*$  be the list of all tiles in the hierarchy: it includes the tiles in  $\mathcal{T}_0$  and all their descendants from the subdivision procedure. The tiles in  $\mathcal{T}_*$  naturally have a forest structure associated with them in which the forest roots are root tiles. Let *leaf tile* denote a triangle in  $\mathcal{T}_*$  that is not further subdivided during the generalized pinwheel subdivision phase. Let  $\mathcal{T}_f$  be the set of leaf tiles.

The first step in converting the tiling to a mesh is to identify for each edge  $e$  of each tile  $T \in \mathcal{T}_*$  the edge

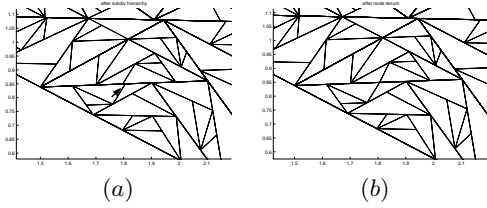
that we denote  $\text{big}(e)$ . This is defined to be the edge  $e'$  of a triangle  $T'$  higher up in the subdivision hierarchy (i.e.,  $T$  is derived from  $T'$  via a sequence of zero or more subdivision operations) such that  $e \subset e'$ , and such that  $e'$  is maximal with this property (i.e., there is no other ancestor of  $T$  with an edge  $e''$  that strictly contains  $e'$ ).

For each triangle in  $\mathcal{T}_0$ ,  $\text{big}(e) = e$ . For some other tile  $T$  with an edge  $e$ , it is a straightforward matter based on a checking a finite number of cases whether  $\text{big}(e) = e$  or  $\text{big}(e) \neq e$ . In the latter case,  $\text{big}(e)$  can be determined from the immediate parent of  $T$  (assuming  $\text{big}(e)$  is already tabulated for the the parent's edges). Thus, it is possible to determine  $\text{big}(e)$  for each edge of each tile in  $\mathcal{T}_*$  with a constant number of operations per tile.

Next, for each “big” edge  $e$  (that is, an edge such that  $\text{big}(e) = e$ ), identify a *moving* and *staying* side. This choice can be quite arbitrary, except for two stipulations. An edge  $e$  adjacent on the exterior boundary of  $\Omega$  should have its inside labeled staying (i.e., no tiles lie on its moving side). An edge in correspondence with  $CF$  in Figure 3 (every big edge generated during the subdivision procedure is in correspondence with either  $DE, DF, CF$  or  $DG$ ) should have the side facing vertex  $B$  labeled as moving. We now identify all the nodes on the staying side of  $e$  and sort them in order of occurrence on the edge. This sorted list is saved for the next phase of the algorithm.

In the next phase, we loop over triangles in  $\mathcal{T}_* - \mathcal{T}_f$  starting from the coarsest and perform collapse-node operations on each. Let  $T$  be a tile in  $\mathcal{T}_* - \mathcal{T}_f$ . Let the four vertices of  $T$  introduced when it is subdivided be labeled  $D, E, F, G$  as in Figure 3. We perform no operation for  $G$  since it is on the staying side of edge  $CF$ . The maximal big edge containing  $D$  is  $\text{big}(AC)$ ; call this  $b(D)$ . The maximal big edge containing  $E$  and  $F$  is  $\text{big}(AB)$ ; call this  $b(E)$  and also call it  $b(F)$ . Let  $v$  be one of  $D, E, F$ . We check whether  $v$  is on the moving side of  $b(v)$ . If it is on the staying side, then no further operation is performed. If it is on the moving side, then we find the vertex  $v'$  taken from the staying side of  $b(v)$  that is closest to  $v$ . This  $v'$  can be found efficiently using binary search on the precomputed sorted lists. If  $\|v - v'\| \leq \delta$ , we collapse nodes  $v$  and  $v'$  by displacing  $v$  to  $v'$ . Here  $\delta$  is a tolerance discussed more below.

This displacement induces uniquely determined affine transformations on triangles contained in  $T$  as follows. If  $v$  is the vertex labeled  $D$  in Figure 3, then there is a unique affine transformation on  $\triangle ADF$  that leaves  $A$  and  $F$  fixed and moves  $D$  to  $v'$ . A second affine transformation of  $\triangle CDF$  leaves  $F$  and  $C$  fixed and moves  $D$  to  $v'$ . If  $v$  is the vertex labeled  $E$ , then there are unique affine transformations determined for



**Figure 6:** Example of a collapse-node operation is shown. A node on one side of a “big” edge that lies within the tolerance of a node on the other side is moved and merged with the nearby node on the other side.

$\triangle ADE$  and  $\triangle DEF$ . Finally, if  $v$  is the vertex labeled  $F$ , then there are transformations for each of  $\triangle DEF$ ,  $\triangle CDF$  and  $\triangle BCF$ . The algorithm applies all the relevant affine transformations caused by motion of the node. Note that the affine transformations agree on the boundaries between these triangles, so there is no consistency issue regarding which transformation to apply. These transformations move the triangle, including every node at deeper levels of the hierarchy contained in it. This concludes the description of the *collapse-node* operation. See Figure 6 for an illustration of this operation.

Note that a single tolerance  $\delta$  is used to determine motion. The theoretical value for  $\delta$  is given by (2) below. We will verify later that this value of  $\delta$  is sufficiently small so that two important properties hold:

**Property 1 of  $\delta$ :** If a vertex  $v'$  is the target of a collapse-node move, then it should be uniquely determined, i.e., there should not be two vertices  $v'$  and  $v''$  on the staying side of  $b(v)$  that are both within distance  $\delta$  of  $v$ .

**Property 2 of  $\delta$ :** No two vertices on the moving side of  $\text{big}(e)$  for any  $e$  should be collapsed to the same node on the staying side.

In a future extension of PINW to handle graded meshes, presumably the value of  $\delta$  would not be a single global value.

The affine transformations described above have the property that all of the segments illustrated in Figure 3 remain straight (collinear) segments after the transformation. It is apparent that each collapse-node operation could cause many nodes to move. We will say that the one node  $v$  that is displaced to match  $v'$  is *directly* displaced. The nodes moved by virtue of an affine transformation induced by moving  $v$  are said to be *indirectly* displaced.

A collapse-node operation, once executed, cannot be undone by future collapse-node operations. The rea-

son is that  $v$  is never moved again. It is never moved again directly since it can be moved directly only when the tile  $T = \triangle ABC$  that created it is processed. It can also never be moved again indirectly since there is no tile in lower levels of the hierarchy that contains it except as a corner vertex, and corner vertices of a triangle  $T'$  are not moved when  $T'$  is processed. Similarly,  $v'$  can never be moved again. The reason is that  $v'$  is never moved directly (since it is on the staying side of its big edge). Any transformation that might move  $v'$  indirectly takes place at a level of the hierarchy higher than the level of  $T$ .

We carry out all available collapse-node operations for all triangles in the order described. Once all collapse-node operations are complete, we are left with the collection of distorted leaf tiles, each of which may have one or more hanging nodes. These hanging nodes are collinear with the endpoints of the edges on which they hang because, as noted above, we do not disturb any collinearity relationships with collapse-node operations. The hanging nodes are all at least  $\delta$  apart from the corners and from each other.

For each of these distorted tiles, we compute its Delaunay triangulation (including the hanging nodes). The collection of all of these Delaunay triangles forms a simplicial mesh that is the final output of PINW.

The running time of PINW is analyzed as follows. Let  $n$  be the number of leaf tiles. Then the total number of tiles is  $O(n)$ , as is the total number of vertices and edges. The heap insertions and deletions require  $O(n \log n)$  total operations. Sorting all the lists associated with big edges requires  $O(n \log n)$  operations. Looking up a vertex in a sorted list requires  $O(\log n)$  operations for binary search, hence all of the lookups to see if a node should be collapsed require  $O(n \log n)$  operations.

The recursive application of affine transformations requires  $O(nd)$  operations since each vertex is transformed at most  $d$  times, where  $d$  is the maximum depth of the forest associated with  $T^*$ . We claim  $d = O(\log n)$ . It follows from Lemmas 3 and 4 in the next section that the minimum altitude of a triangle at depth  $k$  lies between  $\alpha_0 C^k$  and  $\alpha_1 D^k$ , where  $\alpha_0, \alpha_1$  are lower and upper bounds on the minimum altitudes among root tiles,  $D$  is an absolute constant and  $C$  is a scalar depending on the worst aspect ratio among root tiles. This means that a leaf tile can be at most a factor  $\log D / \log C$  (asymptotically) deeper in the forest than any other leaf tile. Thus, all leaves have depth  $O(\log n)$ .

Finally, the Delaunay triangulation operations in the last step of the algorithm also require  $O(n \log n)$  operations total. Overall, we see that PINW requires  $O(n \log n)$  operations.



## 7. ANALYSIS OF ASPECT RATIO

In this section we analyze the aspect ratios of triangles produced by PINW, showing that they are bounded above by a number that depends only on the sharpest angle in the original polygon  $\Omega$ . Before this analysis, we first explain how to select the parameter  $\delta$  described in the last section. The parameter  $\delta$  depends on the minimum altitude of leaf tiles as will be apparent from the theory developed here. Let  $\text{minalt}(T)$  denote the minimum altitude of triangle  $T$ .

**Lemma 2.** *Let  $T$  be a triangle with vertices  $v_1, v_2, v_3$ . Let  $T'$  be the triangle with vertices  $v'_1, v_2, v_3$ . Let  $A$  be the unique affine transformation that carries  $T$  to  $T'$ . Let  $l$  be an arbitrary line segment. Then*

$$1 - d/a \leq \frac{\text{length}(A(l))}{\text{length}(l)} \leq 1 + d/a \quad (1)$$

where  $d = \text{dist}(v_1, v'_1)$  and  $a$  is the altitude of  $v_1$  with respect to  $v_2v_3$ .

This result follows in a fairly straightforward manner from a well-known theorem about how the singular values of a matrix change under perturbation.

**Lemma 3.** *Consider the generalized pinwheel subdivision illustrated in Figure 3 of a triangle  $T$ . Assume that  $b \geq \min(.4, a)$  and  $c - a \geq \min(.4, a)$ . Then letting  $T'$  be any one of the five subtriangles, we have  $\text{minalt}(T') \leq 0.9725 \text{minalt}(T)$ .*

**Remark 1.** The assumptions are valid for all tiles produced by PINW. For root tiles, we have ordered the angles  $a \leq b \leq c$ , and we know  $c - a \geq .4$  because of the preliminary step of splitting near-equilateral triangles. For conjugates of root triangles, say  $a = a'$ ,  $b = c' - a'$ , and  $c = \pi - c'$  where  $a', b', c'$  are the angles of a root tile, we know also  $c - a = \pi - c' - a' = b' \geq a$  and that  $b = c' - a' \geq .4$ .

**Remark 2.** The factor 0.9725 is due to our proof technique and appears to be an overestimate. A search over a fairly dense grid of possible angles satisfying the hypotheses of the theorem indicates that the true bound is closer to 0.918.

*Proof.* Due to space limitations, we merely sketch out the main ideas of the proof. Using the law of sines and the assumptions, we prove many inequalities about the side lengths in Figure 3 such as  $|BC| \leq .922|AB|$ . With enough inequalities like this, we can then argue using principles of similar triangles about the minimum altitude.  $\square$

The following lemma is like the previous one except with an inequality in the opposite direction.

**Lemma 4.** *Consider the generalized pinwheel subdivision illustrated in Figure 3 of a triangle  $T$ . Assume that  $b \geq \min(.4, a)$  and  $c - a \geq \min(.4, a)$ . Then letting  $T'$  be any one of the five subtriangles, we have  $\text{minalt}(T') \geq p \text{minalt}(T)$ , where for subtriangles I, II, III, IV,  $p \geq 0.0044$  and for subtriangle V,  $p \geq \sin a$ .*

**Remark.** The factor 0.0044 is due to our proof technique and appears to be an underestimate. A search over a fairly dense grid of possible angles satisfying the hypotheses of the theorem indicates that the true bound is closer to 0.125.

Due to space limitations, the proof is omitted.

**Lemma 5.** *Let  $s, t$  be positive numbers such that  $s < 1$  and  $t < 1$ , and let  $k$  a positive integer. Then*

$$\prod_{i=0}^{\infty} (1 - st^i)^k \geq 1 - ks/(1 - t)$$

and

$$\prod_{i=0}^{\infty} (1 + st^i)^k \leq \exp(ks/(1 - t)).$$

We now explain how to choose  $\delta$  for the main algorithm. We set it to be

$$\delta = \frac{\min\{\text{minalt}(T) : T \in \mathcal{T}_*\}}{1460}. \quad (2)$$

The minimum altitudes in this definition are measured before any collapse-node operations begin. This choice of  $\delta$  makes all the theorems work but leads to poorer aspect ratio (by a constant factor) than seems necessary. So instead, PINW 1.0 chooses  $\delta$  dynamically based on the singular values of the affine transformations that could be applied during collapse-node operations. This heuristic seems to work well in practice.

The following theorem bounds the effect of all collapse-node operations, both direct and indirect.

**Theorem 5.** *Let  $T$  be a tile in the hierarchy generated by PINW, and let  $A$  be the composition of all the affine transformations applied directly to vertices of  $T$  and indirectly to those vertices via ancestors in the hierarchy. Let  $\alpha = \text{minalt}(T)$  (prior to any node movement). Let  $l$  be a line segment lying in  $T$ . Assume  $\delta$  is chosen according to (2). Then  $\text{length}(A(l))/\text{length}(l)$  lies between*

$$\left(1 - \frac{\delta}{0.75\alpha}\right)^3 \cdot \left(1 - \frac{0.9725\delta}{0.75\alpha}\right)^3 \cdot \left(1 - \frac{0.9725^2\delta}{0.75\alpha}\right)^3 \dots$$

and

$$\left(1 + \frac{\delta}{0.75\alpha}\right)^3 \cdot \left(1 + \frac{0.9725\delta}{0.75\alpha}\right)^3 \cdot \left(1 + \frac{0.9725^2\delta}{0.75\alpha}\right)^3 \dots$$

Due to space limitations, we defer the proof of this theorem to the full paper.

We now consider Properties 1 and 2 in Section 6. Since the minimum altitude of a triangle is less than or equal to its shortest side length, and since the minimum altitude of any tile decreases by at most 0.75, the previous result shows that  $\delta$  is sufficiently small so that no two nodes can be collapsed to the same node, and no node can have more than one choice of where it should be collapsed.

Furthermore, when we are finished with collapse-node operations, all hanging nodes are at least  $\delta$  apart and at least  $\delta$  from corners. Again, this is because the shortest side length is bounded below by the smallest altitude, and the smallest altitude is bounded below by a large constant multiple of  $\delta$ .

We now consider the aspect ratio of the triangles in the mesh produced by PINW. We define the *aspect ratio* of a triangle to be the square of the longest side length of the triangle divided by its area. Since the area is half the product of the longest side length and the minimum altitude, an equivalent definition is twice the longest side length over the minimum altitude.

The first step of PINW, which performs a preliminary triangulation of  $\Omega$  using Triangle, outputs triangles that have their aspect ratios bounded above. The reason is that Triangle is a guaranteed-quality mesh generation algorithm that will put sharp angles into its output only when the input polygon has very sharp angles. Thus, the small angles of all the initial triangles have a lower bound. (The reciprocal of the smallest angle of a triangle is within a constant factor of the aspect ratio definition given in the previous paragraph.) The operation of subdividing at in-centers done to obtain  $\mathcal{T}_0$  from Triangle's output does not increase the longest side length, and reduces the area by at most a constant factor. Hence the triangles in  $\mathcal{T}_0$  still have bounded aspect ratio.

Next, we consider the tiles in  $\mathcal{T}_f$ , that is, the leaf tiles. Each of these is similar to a root tile or its conjugate. In a preliminary step, we ensured that  $c-a$  is bounded below for all conjugates of root tiles. Therefore, the leaf tiles all have bounded aspect ratio.

In more detail, the smallest angle of each conjugate leaf tile is either  $a$ , where  $a$  is the smallest angle of a root tile, or is  $c-a$ , where  $a$  is the smallest and  $c$  is the largest angle of a root tile. But we have ensured that  $c-a > .4$  by our preliminary splitting rule. Thus, if the smallest angle of a conjugate tile is  $c-a$ , this means that the conjugate tile has a universal upper bound on its aspect ratio.

Now, we consider the effect of collapse-node operations.

**Lemma 6.** *After all collapse-node operations are complete, the aspect ratio of any leaf tile has increased (compared to its value prior to all collapse-node operations) by at most a factor of 1.22.*

For this theorem and the remainder of the section, let  $R_1$  denote the maximum aspect ratio among root tiles and their conjugates. As noted above, because of the properties of Triangle,  $R_1$  is bounded above by a constant multiple of the reciprocal of the sharpest angle of  $\Omega$ .

**Theorem 6.** *Assume that no root tile is in  $\mathcal{T}_f$  (i.e., each triangle in  $\mathcal{T}_0$  is split at least once by the PINW subdivision procedure). Then, prior to collapse-node operations, the maximum value of the minimum altitude among all leaf tiles is no more than  $cR_1$  times the minimum value of the minimum altitude among all leaf tiles, where  $c$  is a universal constant.*

*Proof.* Recall that the tile selected for splitting at any given step is the one with the maximum minimum altitude. Thus, when the subdivision procedure terminates, the tile at the top of the heap will be the leaf tile with the maximum minimum altitude among all leaf tiles. Say this tile is  $T$  and its minimum altitude is  $\alpha$ . Now consider any other leaf tile  $T' \in \mathcal{T}_f$ . Because of the assumption that no tile from  $\mathcal{T}_0$  is a leaf tile, this tile  $T'$  must have arisen from a subdivision of some other tile  $T''$ . Because of the heap order, the minimum altitude  $\alpha''$  of  $T''$  exceeds  $\alpha$ . By Lemma 4, this means that the minimum altitude  $\alpha'$  of  $T'$  is at least  $\min(0.0044, \sin a)\alpha''$ . Note that  $\sin a \geq c/R_1$  since  $a$  is an angle of a root tile. Thus,  $\alpha' \geq (c/R_1)\alpha'' \geq (c/R_1)\alpha$ .  $\square$

We now come to the main result for this section about the aspect ratio of the triangles generated by PINW.

**Theorem 7.** *Each triangle in the simplicial mesh output by PINW has aspect ratio at most  $cR_1^3$ , where  $c$  is a universal constant and  $R_1$  was defined above to be the largest aspect ratio among root tiles.*

Due to space limitations, we omit the proof. In brief, the idea is that we consider the sharpest angle of a triangle  $\tau$  in the Delaunay triangulation. There are two cases: either the edge  $e = v_1v_2$  opposite the sharp angle lies on an edge  $E$  of  $T$  or it connects two edges of  $T$ . Consider only the first case. We can bound the sharp angle of  $\tau$  by the sharp angle  $v_1vv_2$  of the triangle  $\tau'$  resulting from joining  $e$  to  $v$ , where  $v$  is the vertex of  $T$  opposite  $E$ . We find that the aspect ratio of  $\tau'$  is proportional to  $R_1^3$ . One factor of  $R_1$  arises because  $|e|/\text{diam}(T) \geq c/R_1$ . The other two factors of  $R_1$  arise because if  $R_1$  is large,  $v$  could make a large angle (close to  $\pi$ ) with  $e$ .

## 8. ISOPERIMETRY OF THE FINAL MESH

We have already proved in Theorem 3 that the tiling of a triangle by our generalized pinwheel subdivision has the isoperimetric property. It is straightforward to extend this result to the collection of all leaf tiles.

**Theorem 8.** *Let  $\mathcal{T}_0, \mathcal{T}_1, \dots$ , be the sequence of tilings of  $\Omega$  generated by the PINW algorithm as follows. For each  $n$ ,  $\mathcal{T}_n$  is the set of leaf tiles of  $\Omega$  generated by PINW when the user-specified size requirement is  $\delta_n$  such that  $\delta_n \rightarrow 0$  as  $n \rightarrow \infty$ . Then for any distinct points  $P, Q$  such that  $P, Q \in \text{Skel}(\mathcal{T}_k)$  for some  $k$ , we have*

$$\lim_{\substack{n \rightarrow \infty \\ n \geq k}} \text{dist}_{\text{Skel}(\mathcal{T}_n)}(P, Q) = \|P - Q\|.$$

*Proof.* This theorem follows from Theorem 4 and uses the same proof technique. In particular, for each tile  $T_i$  in  $\mathcal{T}_k$  that meets  $PQ$ , consider the small segment  $P_iQ_i$  that is  $T_i \cap PQ$ . Then we use Theorem 3 to argue that this small segment  $P_iQ_i$  can be approximated arbitrarily accurately.  $\square$

This theorem can now be extended to the final mesh output by PINW by analyzing the effect of collapse-node operations on the isoperimetric number. (The Delaunay operations do not disturb the isoperimetry result, since adding edges could only make the isoperimetric number decrease.)

The definition of isoperimetry implicit in Theorems 4 and 8 is not suitable for analyzing the output of PINW because the meshes produced by PINW are not refinements of their predecessors as the mesh size decreases. This is because the collapse-node operations move nodes differently depending on the size of the leaf tiles.

Therefore, we use the following definition. An infinite sequence of simplicial meshes  $\mathcal{M}_1, \mathcal{M}_2, \dots$  for a domain  $\Omega$  has the *isoperimetric property* if for each  $\mathcal{M}_i$  there is a subset  $L_i$  of its vertices such that the following two properties hold. First,  $L_i$  is asymptotically dense in  $\Omega$  as  $i \rightarrow \infty$ , i.e., for any  $\epsilon > 0$ , there is an  $I$  such that for any  $x \in \Omega$  and any  $i > I$ , there exists a  $v \in L_i$  such that  $\|x - v\| \leq \epsilon$ . Second,

$$\limsup_{i \rightarrow \infty} \left\{ \frac{\text{dist}_{\text{Skel}(\mathcal{M}_i)}(x, y)}{\|x - y\|} : x, y \in L_i; x \neq y \right\} = 1.$$

**Theorem 9.** *The family of meshes produced by the PINW algorithm has the isoperimetry property described in the previous paragraph.*

*Proof.* To show that PINW has this property, take a sequence of  $\epsilon_i$ 's tending to zero. For each  $i$ , let  $\mathcal{T}_i$

be a generalized pinwheel subdivision of  $\Omega$  such that each leaf cell has diameter less than  $\epsilon_i/2$ . Then let  $\mathcal{T}'_i$  be a further subdivision of  $\mathcal{T}_i$  such that for any two distinct vertices of  $\mathcal{T}_i$ ,  $\text{dist}_{\text{Skel}(\mathcal{T}'_i)}(x - y) \leq (1 + \epsilon_i/4)\|x - y\|$ . The existence of such an  $\mathcal{T}'_i$  is established by Theorem 8. Let  $\alpha'$  be the minimum altitude among leaf tiles in  $\mathcal{T}'_i$ . Next, further refine  $\mathcal{T}'_i$  to yield a tiling  $\mathcal{T}''_i$  with the property that when  $\delta$  is defined by (2) for  $\mathcal{T}''_i$ , (i.e., the  $\mathcal{T}_*$  appearing in (2) pertains to  $\mathcal{T}''_i$ ), then this  $\delta$  is sufficiently small so that

$$\exp(3\delta/(0.75\alpha'(1 - 0.9725))) \leq 1 + \epsilon_i/4 \quad (3)$$

and

$$1 - \epsilon_i/4 \leq 1 - 3\delta/(0.75\alpha'(1 - 0.9725)). \quad (4)$$

Now finally, take  $\mathcal{M}_i$  to be the simplicial mesh output by PINW based on  $\mathcal{T}''_i$ , and take  $L_i$  to be the set of nodes of  $\mathcal{M}_i$  that are displaced copies of the nodes of  $\mathcal{T}_i$ .

First, we have to show that  $L_i$  defined in this manner is asymptotically dense. The nodes of  $L_i$  are the same as the nodes of  $\mathcal{T}_i$  after small displacements. Since every cell of  $\mathcal{T}_i$  has diameter less than  $\epsilon_i/2$ , this means that any point  $x \in \Omega$  is distance at most  $\epsilon_i/2$  from a vertex of  $\mathcal{T}_i$ . The vertices of  $L_i$  are slightly displaced, but no distance  $d$  decreases below  $0.9d$  nor increases to more than  $1.09d$ . Therefore, for any  $x \in \Omega$  the perturbed set  $L_i$  contains a point  $v$  within distance  $1.09 \cdot \epsilon_i/2 < \epsilon_i$  of  $x$ .

Let  $x, y$  be two distinct points in  $L_i$ . The next task is to show that  $\text{dist}_{\text{Skel}(\mathcal{M}_i)}(x, y) \leq \|x - y\|(1 + \epsilon_i)$ . Let  $x_0, y_0$  be the positions of  $x, y$  in  $\mathcal{T}'_i$  prior to all distortions caused by collapse-node operations. Note that  $x_0, y_0$  are vertices of  $\mathcal{T}'_i$  and also of  $\mathcal{T}_i$  by construction. Therefore, by construction of  $\mathcal{T}'_i$ , there is a path  $P_0$  in  $\text{Skel}(\mathcal{T}'_i)$  connecting  $x_0$  and  $y_0$  such that  $\text{length}(P_0) \leq \|x_0 - y_0\| \cdot (1 + \epsilon_i/4)$ . Let the segments of  $P_0$  be  $e_1, e_2, \dots, e_r$ . Let the image of  $P_0$  after all the collapse-node operations with their attendant distortions are applied be  $P$ , and the images of  $e_1, \dots, e_r$  be  $f_1, \dots, f_r$ . Recall that the distortions that affect a node  $v$  of a tile  $T$  in the hierarchy are those distortions associated with  $T$  and its ancestor tiles, but descendant tiles cannot move  $T$ . Therefore, by Theorem 5, all of the quantities  $\|x - y\|/\|x_0 - y_0\|$ ,  $\text{length}(f_i)/\text{length}(e_i)$ , and  $\text{length}(P)/\text{length}(P_0)$  lie between

$$\left(1 - \frac{\delta}{0.75\alpha'}\right)^3 \cdot \left(1 - \frac{0.9725\delta}{0.75\alpha'}\right)^3 \cdot \left(1 - \frac{0.9725^2\delta}{0.75\alpha'}\right)^3 \dots$$

and

$$\left(1 + \frac{\delta}{0.75\alpha'}\right)^3 \cdot \left(1 + \frac{0.9725\delta}{0.75\alpha'}\right)^3 \cdot \left(1 + \frac{0.9725^2\delta}{0.75\alpha'}\right)^3 \dots$$

where the  $\delta$  in this formula is given by (2) associated with  $\mathcal{T}_i''$ . By Lemma 5, this interval is bracketed by

$$1 - 3 \frac{\delta}{0.75\alpha'(1 - 0.9725)}$$

and

$$\exp(3\delta/(0.75\alpha'(1 - 0.9725))).$$

Then by (3) and (4), this interval is bracketed by  $[1 - \epsilon_i/4, 1 + \epsilon_i/4]$ . Thus,

$$\begin{aligned} \text{dist}_{\text{Skel}(\mathcal{M}_i)}(x, y) &\leq (1 + \epsilon_i/4) \text{dist}_{\text{Skel}(\mathcal{T}_i')}(x_0, y_0) \\ &\leq (1 + \epsilon_i/4)^2 \|x_0 - y_0\| \\ &\leq (1 + \epsilon_i/4)^2 \|x - y\| / (1 - \epsilon_i/4). \end{aligned}$$

Note that  $(1 + \epsilon_i/4)^2 / (1 - \epsilon_i/4) \leq 1 + \epsilon_i$  as long as  $\epsilon_i \leq 1/2$ . Thus, we have shown that for all  $x, y \in L_i$ ,  $\text{dist}_{\text{Skel}(\mathcal{M}_i)}(x, y) \leq (1 + \epsilon_i) \|x - y\|$ .  $\square$

## 9. CONCLUSIONS

We believe that this generalization of pinwheel tiling to meshing polygonal regions would aid in modeling arbitrary crack paths more accurately than the current meshing techniques. This work raises a number of interesting directions for future research:

1. The transformation of the tiling to the mesh had the effect of increasing the aspect ratio significantly. Is there a better way to carry out this transformation to reduce the impact on aspect ratio?
2. The convergence rate of the isoperimetric number of the pinwheel tiling to 1, which was not analyzed here, is known to be extremely slow even in the case of the 1:2 tiling. Is there another approach to isoperimetry that converges faster?
3. Consider a mesh generated by placing random points in the domain under consideration and joining them with a Delaunay triangulation. Is there a limiting isoperimetric number for this family of meshes (with high probability)?
4. Another way to construct a mesh of an arbitrary polygon with limiting isoperimetric number equal to 1 is to use the 1:2 pinwheel subdivision for every coarse triangle after subjecting it to a (potentially large) affine transformation. This approach is simpler in certain respects than PINW. For example, the collapse-node operations for this algorithm need to be done only at the boundaries of the coarse triangles. The difficulty with this approach is that it spoils the “statistical rotational invariance” of the pinwheel tiling. The statistical rotational invariance property states that the set

of possible directions is covered at a uniform rate as subdivision proceeds. We are unclear whether statistical rotational invariance is important for cohesive interface modeling. We suspect that our construction of generalized pinwheels has statistical rotational invariance but have no proof of this.

5. Can any of this work be extended to three dimensions?

## 10. ACKNOWLEDGEMENTS

We are grateful to Marshall Bern for originally telling us about Radin’s paper. We also thank the two reviewers of the short version of this paper submitted to the 2004 International Meshing Roundtable for their helpful comments.

## References

- [1] Papoulia K.D., Sam C.H., Vavasis S. “Time continuity in cohesive finite element modeling.” *International Journal for Numerical Methods in Engineering*, vol. 58, no. 5, 679–701, 2003
- [2] Brenner S.C., Scott L.R. *The mathematical theory of finite element methods*. Springer, New York, 1994
- [3] Radin C. “The Pinwheel Tilings of the Plane.” *The Annals of Mathematics*, vol. 139, no. 3, 661–702, 1994
- [4] Radin C., Sadun L. “The Isoperimetric Problem for Pinwheel Tilings.” *Communication in Mathematical Physics*, vol. 177, 255–263, 1996
- [5] Xu X.P., Needleman A. “Numerical Simulations of Fast Crack Growth in Brittle Solids.” *Journal of Mechanics and Physics of Solids*, vol. 42, no. 9, 1397–1434, 1994
- [6] Sadun L. “Some Generalizations of the Pinwheel Tilings.” *Discrete and Computational Geometry*, vol. 20, 79–110, 1998
- [7] Shewchuk J.R. “Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator.” M.C. Lin, D. Manocha, editors, *Applied Computational Geometry: Towards Geometric Engineering*, vol. 1148 of *Lecture Notes in Computer Science*, pp. 203–222. Springer-Verlag, May 1996. From the First ACM Workshop on Applied Computational Geometry
- [8] Aho A.V., Hopcroft J.E., Ullman J.D. *Data structures and algorithms*. Addison-Wesley, Reading, Mass., 1983