Square Tilings and Discrete Modulus

Honors Thesis (HONR 499)

by

Andy Reel

Thesis Advisor

Dr. Guy C. David

Ball State University
Muncie, Indiana

December 2019

Expected Date of Graduation

May 2020

Abstract

In his 1993 paper, "Square Tilings with Prescribed Combinatorics", Oded Schramm gave a remarkable one-to-one correspondence between triangulations of planar regions and tilings of rectangles by squares. Schramm uses a discrete version of a notion known as "extremal length" to describe his ideas and formulate his proofs. In this paper, the same ideas are explored but using "modulus," the inverse of extremal length. In doing so, a new perspective is introduced on how to understand the connection between triangulations and square tilings. Pictures and examples are included to help illustrate just how the use of modulus makes these ideas more accessible and more easily understood. These examples were created through the use of Mathematica code, and several potential applications are discussed. Schramm's proofs are also clarified.

Acknowledgements

I would like to thank Dr. Guy C. David, who served as my advisor over the last seven months. His guidance and patience, demonstrated through weekly meetings during Spring, Summer, and Fall 2019, have been immeasurably beneficial.

I would also like to thank Dr. Rebecca Pierce for permitting me to present my findings through Ball State University's Undergraduate Pursuits Colloquium Series, as well as Dr. Kathryn Shafer for attending my presentation at the Indiana Undergraduate Math Research Conference in July 2019. The support from Ball State's Department of Mathematics has been encouraging and appreciated

Process Analysis Statement

As I approached the conclusion of my undergraduate experience as a Secondary Education, Mathematics major, I found myself seeking further exploration into the world of mathematics. The decision to research mathematics was two-fold: I sought a personal challenge as a way to test my abilities as a mathematician, and I wanted to build my own mathematical knowledge in order to better prepare myself for a potential pursuit of a graduate degree. This particular topic was a joint decision between Dr. Guy David, my advisor, and I. Dr. David proposed several topics, and the concept of Square Tilings and Discrete Modulus interested me the most. Drawing from the concept of circle packing, which is used in modern medicine and MRI imaging, square tiling is a relatively recent discovery that has a lot to be discovered. The potential for real-world application was a strong influence in choosing this topic.

While the theory discussed in my paper seems inaccessible to those without a mathematics background, a major goal of this paper is to make the ideas discussed by previous mathematicians understandable to the lay academic. Each section is supplemented with examples or explanations that break down the content previously discussed. My hope is that anyone can read my thesis and gain an understanding of the relationship between triangulations and square tilings as well as an appreciation of the resulting square tilings and examples, which have never been produced before.

The primary text on square tilings is titled “Square Tilings with Prescribed Combinatorics,” which was authored by Oded Schramm and published in 1993. My research began with this text, as my first objective was to understand exactly what I would be researching for the next several months. For the bulk of April 2019, I met weekly with Dr. David to discuss this paper. This meetings consisted of detailed explanations and discussions of the theorems and proofs that Schramm provided. In his paper, Schramm employs the notion of “discrete extremal length” to make calculations necessary for his proofs. Once I understood his application of discrete extremal length, I set about reworking his ideas and theorems. However, instead of using discrete extremal length, I employed “discrete modulus,” which is the inverse of discrete extremal length.

Over the summer, I continued to meet with Dr. David once a week at a coffee shop in Carmel. I found this to be extremely helpful, as it provided me with the motivation necessary to stay on task. My objective for the summer was to rework and reprove Schramm’s theorems using discrete modulus. I ran into several issues during this stage. Repeatedly, I would become frustrated due to a lack of understanding; the weekly meetings often consisted of reiteration of concepts so I could prove the theorems with ample explanation and rigor. It was difficult to admit defeat, and it became increasingly disheartening as the summer bore on and I continued to make mistakes. In addition to mathematical mishaps, I also had to learn how to type in LaTeX, which is a software that takes text commands and transforms them into mathematical language and symbols. There were several technological issues I encountered. I had no LateX software, so I had to research and choose the correct one and learn the coding language independently. I also had no way of using the language that was displayed, as I was unaware of existing software that incorporates LaTeX text into research papers. As a result, I screenshotted each and every line and dragged the JPEG files into a Google Doc. Dr. David seemed thoroughly amused at my struggle.

By the end of the summer, I had found myself proficient in LaTeX and discrete modulus, having reworked and reporved all the theorems by the start of the Fall 2019 semester. At this stage, it became apparent that I had spent all summer discussing and writing about square tilings, but I had yet to produce any using Schramm's provided algorithm. Schramm actually provides very few examples of square tilings in his paper, as running the algorithm using paper and pencil can be immensely time-consuming or downright impossible. After some trivial, hand-drawn examples, I began tackling my next objective: writing a Mathematica code that took triangulations (graphs comprised of triangles) and transformed them into square tilings using the theory I proved over the summer. This seemed insurmountable at the time. I had experience writing proofs and pondering challenging mathematical concepts from my math classes, but I had no knowledge of code besides my recently acquired LaTeX skills. Unfortunately, Mathematica code is not the same as LaTeX, so I had to start the process over again. Over the next two months, I made various attempts at writing a working code from the ground-up, taking into consideration both the applicability and accuracy as well as the user experience. There were a multitude of roadblocks I ran into as I waded through lines and lines of confusing code, and some of them were more easily fixed than others.

Once I had finished the first round of code, and I began generating trivial square tilings, I was ecstatic. However, as I ventured into more difficult tilings, my code stopped performing correctly. I was devastated; the thought of rewriting all of this code again was terrifying. After about a week of toiling, I had discovered the issue. The code takes triangulations, turns them into directed graphs, and finds the "shortest path" from one side to another by using numbers assigned to vertices. However, since Mathematica doesn't understand the concept of "sides," I had to introduce "phantom vertices" to the graph to represent the four pieces of the boundary. These vertices are always assigned 0, so they have no weight in the code. However, I didn't consider that the code may use these phantom vertices in finding the shortest path. In order to fix it, I had to be careful about the directional edges that connect the graph. By limiting paths in and out of the phantom vertices, the code ran smoothly once more.

The next big issue I encountered was when I began triangulating continents in order to come up with examples. Creating a square tiling of Europe seemed like a great way to display exactly what the algorithm does. I began by placing a vertex for each country, then connecting two vertices when their corresponding countries border. This yielded several unforeseen issues that prohibited the resulting graph from being a triangulation. Whenever there are islands, countries with only one border, countries that are entirely surrounded by either one or two countries, or large bodies of water that prohibit the continent from being a topological disk, the resulting graph is not a triangulation. I attempted to prove some of these facts, but I was unable to come up with comprehensive arguments. Thus, I simply avoided the issues when I created the triangulation.

Once I had come up with some descriptive examples, I began working with a special kind of triangulation known as a Delaunay Triangulation. Delaunay Triangulations are created from the contacts graphs of Voronoi diagrams, which are generated using random points generated in a topological disk. The inherently random aspect of these triangulations required a rewrite of the code used previously; before, the user could manually input the triangulation, but now the code has to be able to create a triangulation and determine appropriate boundaries. Once I had found a way to generate points in a disk and attain a list of points on the boundary, I had to split up the boundary points into four lists. This was extremely complicated, and the only way I could solve the issue was to ensure that the fourth portion of the boundary only contained two vertices. This interferes slightly with the random aspect of these triangulations, but the resulting square tilings are fascinating nonetheless.

In Schramm's paper, the potential for conformal mappings are briefly discussed. These conformal mappings are transformations that maintain local angles. Schramm states that using a hexagonal mesh does not yield a conformal mapping. However, I explored the Schwarz-Christoffel mapping from the unit circle to a 1x1 square. My hope was that square tilings would provide an approximation of the Schwarz-Christoffel mapping. After generating a triangulation of the unit circle with 161 vertices, I applied a conformal mapping to the triangulation. I also used my Mathematica code to generate a square tiling, comparing the eventual location of vertices in the conformal map to the location of the top-right corner of the corresponding square in the resulting tiling. The most prominent issues were sorting the vertices in order to be compared, which I was able to achieve through some clever Mathematica sorting commands. With only 161 vertices, though, I did not achieve a decent approximation. I hypothesized that more vertices would yield a better approximation; with more squares in the resulting tiling, the squares themselves would be smaller and eventually converge to points at the desired location. However, when attempting to run the same code with 481 vertices, I did not achieve any usable results. After several hours of letting the program run, I was left with an unfinished tiling. Thus, the relationship between square tilings and Schwarz-Christoffel mappings remains an open question.

As aforementioned, I ran into several major obstacles over the past 6 months. However, through my struggles, I made some worthwhile self-discoveries. This project forced my to work on and write extensive code, which I had never done before. Not only did I make significant progress in my skills, but I discovered how much I enjoy coding. Working in Mathematica was certainly the most challenging and fulfilling aspect of this process. In addition, I gained a lot of insight into what graduate studies in mathematics would consist of, and my excitement about taking on difficult mathematical topics and theories has forced me to reconsider my career path with a potential return to academia.

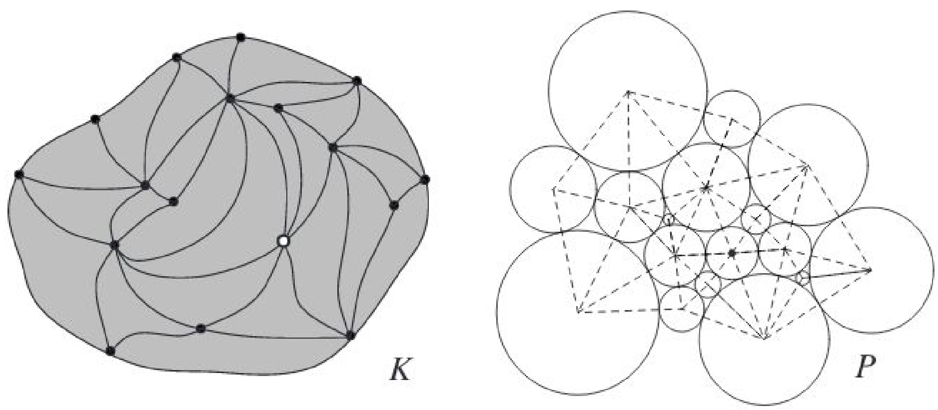
Working with Dr. David and regularly discussing mathematics at length changed the way I perceived learning mathematics. As a secondary education major, I have spent a great deal of time in math lectures and classrooms. Through the Honors College Curriculum, I had experienced discussion-based humanities classes, but I had yet to see such learning applied to mathematics. However, my experience in working through these proofs and problems has altered how I see my role as an educator. The role of a mathematics educator is more than that of an omnipotent lecturer who imparts wisdom upon his students; rather, his role is to facilitate critical thinking and problem-solving through both practice and discussion. Encouraging conversations and discussions about the homework, while often seen as a form of cheating or a distracting practice, can be immensely beneficial. There were many times that Dr. David and I bounced ideas off each other in order to solve a bug in the code, and without this collaboration, my thesis would not have been nearly as fulfilling.

As aforementioned, circle packing is already established as having many important applications. Square packing, however, has not yet been found to have such applications. It is my hope that, once more is understood about the relationship between square tilings and conformal maps or Delaunay Triangulations, more useful applications can be discovered. Besides the theoretical, it may be an interesting experiment to apply the algorithm to triangulations of road maps. The resulting square tilings may yield information about busy intersections or potential areas of traffic without having to collect any actual data. Nevertheless, the potential applications of square tilings were not the focus of my research. Providing a more thorough explanation of Schramm's ideas and developing an accessible code for creating square tilings were the main goals, and both were achieved.

Square Tilings and Discrete Modulus

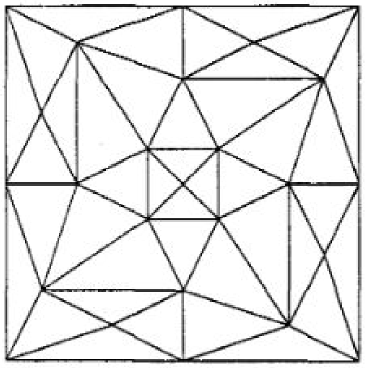
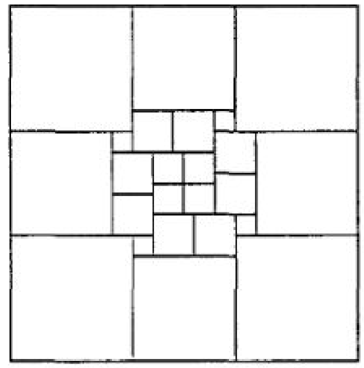
Introduction

The idea of "packing" is, at its base elements, quite simple. Essentially, a packing is an arrangement of shapes inside a larger area. Mathematically speaking, a packing refers to filling a topological disk with a single shape in some prescribed manner. This is usually done using functions that take graphs and sends them to packings. A common example of this is circle packing. In circle packing, the vertices in a graph become circles, and whenever to vertices are connected by an edge, their corresponding circles touch tangentially. Below is an example of a graph and its corresponding circle packing, found in "Circle Packing: A Mathematical Tale" by Kenneth Stephenson [**1**]:



Clearly, the circles must be of different sizes to accommodate the demands of contact, and there exist rules that dictate how large each circle should be. Notice also that the graph K is a triangulation; while circle packings do not demand that their corresponding graphs be triangulations, some types of packings do.

This brings us to the focus of this paper: square packings, or square tilings. These packings are special in a couple ways. First of all, the packings that we will consider are always in the form of rectangles filled precisely with squares, hence the name "tilings." In other words, the squares perfectly fit inside the rectangle with no overlapping, gaps, or spills outside the boundary. Furthermore, the corresponding graphs must be triangulations. Below is an example of a square tiling from "Square Tilings with Prescribed Combinatorics" by Oded Schramm [**2**]:

In these square packings, more than just contacts will be preserved; if a vertex lies on the boundary of a graph, then the corresponding square will lie on the boundary of its rectangle. Moreover, if a vertex is on a "corner" in the graph (this will be explained in more precise detail below), then the corresponding square will lie in the corner of its rectangle. This peculiar and remarkable relationship was first discovered and proven by Schramm in his aforementioned paper. Schramm also provides an algorithm for computing the sizes of the squares in the tiling. In this paper, his methods are clarified and expanded upon.

In order to begin a precise discussion of square packings, we must first define important terms. Let us define a **graph** G=(V,E), where V is the set of vertices and E is the set of edges (denoted by pairs of vertices). We consider "contact" between two vertices to exist when they are connected by an edge. All graphs discussed here will be **connected** graphs. If each pair of vertices is connected by some path of edges, then the graph is considered connected.

The set up for Schramm's theorem is as follows: Let D be a closed triangulated topological disk, and let the set of vertices, edges, and faces of the triangulation be V, E, and F, respectively. Let $\partial D = B_1 \cup B_2 \cup B_3 \cup B_4$ be a decomposition of the boundary of the triangulation into 4 nontrivial arcs, in cyclic (clockwise) order. That is, each B_j is a nonempty connected union of edges of the triangulation, and $B_1 \cap B_3 = \varnothing$ and $B_2 \cap B_4 = \varnothing$. The collection T = $(V,E,F;B_1,B_2,B_3,B_4)$ is called a **triangulation of a quadrilateral**.

Having defined a triangulation, Schramm's Theorem [**2**] can be stated.

Schramm's Theorem: Let $T=(V,E,F;B_1,B_2,B_3,B_4)$ be a triangulation of a quadrilateral. Then there is an $h > 0$ and a square tiling $Z = (Z_v : v \in V)$ of the rectangle $R = [0,h^{-1}] \times [0,h]$ such that

(1) $Z_v \cap Z_u \neq \varnothing$ whenever $\langle v,u \rangle \in E$, i.e., whenever two vertices v and u are connected by an edge.

Moreover, let R_1, R_2, R_3, R_4 be the bottom, left, top, and right edges of the rectangle R, respectively. Then, it is required that for each j = 1,2,3,4, we have:

(2) $Z_v \cap R_j \neq \varnothing$ whenever $v \in B_j$, i.e., whenever the vertex is on the boundary of T.

Under these conditions, the number h and the tiling Z are uniquely determined.

Note that, given this theorem (called **Theorem 1.3** in [**2**]), some squares may degenerate to points, but their contacts will be preserved. This occurs because squares are not smooth.

The proof of this theorem, as given by Schramm, uses an idea called discrete extremal length. We will instead use discrete modulus, the inverse of discrete extremal length, to prove **Theorem 3** and **Lemma 4**, which are both stated in a later section. Modulus is a more commonly used notion in present-day mathematics. Modulus can also be understood through “masses” and the assignment of “weights” to individual vertices. Such terms refer to the relative “importance” of each vertex in the scope of the graph’s connectedness. This will be further explained in the following section.

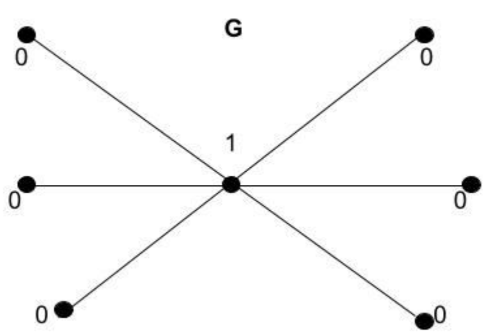
Discrete Modulus

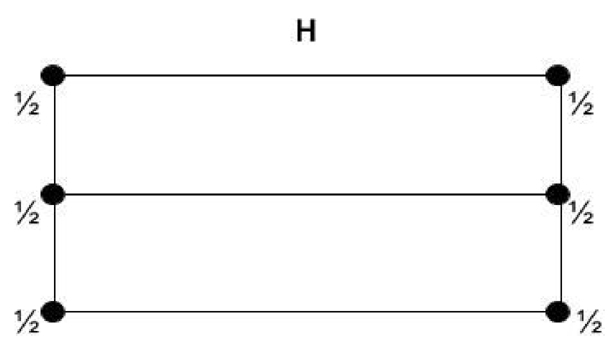
Let G = (V,E) be a finite connected graph. Consider two subsets of vertices, *A* and *B*. Consider now all the paths connecting *A* and *B*; call this a **family of paths,** notated by (G, *A, B*). Now, take a function m:V→[0,∞) such that for each path in the family, $\sum m(v_i) \geq 1$, where v_i are all the vertices on the path. We call m **admissible** if the above holds. Then, inf{ $\sum m(v)^2$ for all vertices : m is admissible } = M, where M is the **modulus** of the family of paths, notated by M(G, *A, B*).

This leads us to the following lemma:

Lemma 2: There exists a unique function m:V→[0,∞) such that $\sum m(v)^2$ over all vertices is equal to inf{$\sum m(v)^2$} = M. [**2**]

Such a concept may be better understood through pictures. Consider the following graphs, G and H:





The assignment of m-values to each vertex is optimal for each graph, i.e., these assignments will yield the best, or smallest, modulus. Let A_G, B_G and A_H, B_H represent the subsets of vertices, respectively, on the left and right of each of the graphs. Any path ***α*** from *A* to *B* yields an admissible m for both graphs (0 + 1 + 0 $\geq$ 1, and ½ +½ $\geq$ 1). However, the family of paths from A_H to B_H is more “well-connected” than the family of paths from A_G to B_G, and modulus is essentially a measure of “connectedness”. Calculating the modulus of each family of paths shows that M(G, A_G, B_G) < M(H, A_H, B_H)

$$M(\mathrm{G}, A_G, B_G) = \sum_{v \in \alpha_G} m(v_i)^2 = 6 \cdot (0^2) + 1^2 = 1 < M(\mathrm{H}, A_H, B_H) = \sum_{v \in \alpha_H} m(v_i)^2 = 6 \cdot (\frac{1}{2})^2 = \frac{3}{2}$$

Thus, with a larger modulus, it is clear that the family of paths from A_H to B_H is more "well-connected" than the family of paths from A_G to B_G. Generally speaking, if there are n vertices on either side of graph H, then the modulus will be n/2. However, no matter how many vertices are on either side of G, the modulus will still be 1.

In addition, the value of $\sum m(v)^2$ is sometimes referred to as the "mass of m" because it is the total squares of all the "weights," or m-values, on a given path. Notice that the middle vertex in G has all of the weight; this is because no path from A_G to B_G can exist without passing through this vertex, so this vertex is essential to the connectedness of the path family. Thus, it has a higher m-value. In graph H, all vertices are equally important, hence their equal weights.

Proof of Lemma 2:

First, we must prove the existence of a function m:V→[0,∞) such that $\sum m(v)^2$ over all vertices is equal to $\inf\{\sum m(v)^2\}$ = M. Consider a graph G, and let G have N vertices $v_1, v_2, v_3, \ldots v_n$. Consider also any metric m as a point in $\mathbb{R}^n$ with coordinates ($m(v_1)$, $m(v_2)$,...$m(v_n)$). Because M is supposedly the infimum of the set, all metrics must have mass greater than or equal to M. Thus, we can find metrics m_1, m_2, m_3,... such that, by definition of infimum,

$$M \leq \sum_{\text{all v}} m_n(v)^2 \leq M + \frac{1}{n} \leq M + 1$$

Thus, the metrics are contained in the bounded set B={x: √(M ≤ d(0,x) ≤ √(M+1)}, where d(0,x) indicates the distance of a point, or metric, from the origin in $\mathbb{R}^n$. This set is closed as well. Thus, B is compact. Thus, there exists a convergent subsequence with its limit in B:

$$(m_{n_1}, m_{n_2} m_{n_3}, \ldots) \to m \in B$$

Here, m is also a metric. We can verify its admissibility on some path $\boldsymbol{\alpha}$:

$$\sum_{v \in \alpha} m(v) = \lim_{k\to\infty} \sum_{v \in \alpha} m_{n_k}(v) \geq 1$$

since each $m_{n(k)}$ is admissible. Now, we need to show that $\sum m(v)^2$ = M. We know that

$$M \leq \sum_{\text{all } v} m(v)^2 = \lim_{k\to\infty} \sum_{\text{all } v} m_{n_k}(v)^2 \leq \lim_{k\to\infty} \left(M + \frac{1}{n_k}\right)$$

Thus, by Squeeze Theorem, $\sum m(v)^2$ = M.

Now, we must prove that this metric m is unique. Consider a path γ from B_2 to B_4, and suppose we have two admissible metrics on that path, m_1 and m_2, such that

$$\sum_{\text{v} \in \gamma} m_1(v)^2 = M = \sum_{\text{v} \in \gamma} m_2(v)^2$$

Because these metrics are admissible, their masses must be positive. Let $w(v) = \frac{1}{2}m_1(v) + \frac{1}{2}m_2(v)$. So,

$$\sum_{v\in\gamma} w(v) = \sum_{v\in\gamma}(\frac{1}{2}m_1(v) + \frac{1}{2}m_2(v)) = \frac{1}{2}[\sum_{v\in\gamma} m_1(v) + \sum_{v\in\gamma} m_2(v)]$$

where $m_1(v)$ and $m_2(v)$ are both admissible. So,

$$\frac{1}{2}[\sum_{v\in\gamma} m_1(v) + \sum_{v\in\gamma} m_2(v)] \geq \frac{1}{2}[1+1] \geq 1$$

So, w is admissible. Now, consider the mass of this metric w, given by $\sum w(v)^2$.

$$\sum_{v\in\gamma} w(v)^2 = \sum_{v\in\gamma}(\frac{1}{2}m_1(v) + \frac{1}{2}m_2(v))^2$$

$$= \sum_{v\in\gamma} \frac{1}{4}m_1(v)^2 + \sum_{v\in\gamma} \frac{1}{2}m_1(v)m_2(v) + \sum_{v\in\gamma} \frac{1}{4}m_2(v)^2$$

$$= \frac{M}{4} + \frac{1}{2}\sum_{v\in\gamma} m_1(v)m_2(v) + \frac{M}{4}$$

$$= \frac{M}{2} + \frac{1}{2}\sum_{v\in\gamma} m_1(v)m_2(v)$$

$$\leq \frac{M}{2} + \frac{1}{2}\cdot\left[\left(\sum_{v\in\gamma} m_1(v)^2\right)^{\frac{1}{2}} \cdot \left(\sum_{v\in\gamma} m_2(v)^2\right)^{\frac{1}{2}}\right] \text{ (by the Cauchy-Schwarz Inequality)}$$

$$= \frac{M}{2} + \frac{1}{2}\left[\sqrt{M}\cdot\sqrt{M}\right] = M$$

So, $\sum w(v)^2 \leq M$, but because M is the infimum of the set of all $\sum m(v)^2$ such that m is an admissible metric, $\sum w(v)^2$ cannot be less than M. Thus, $\sum w(v)^2 = M$. This means that, actually,

$$\sum_{v\in\gamma} m_1(v)m_2(v) = \left[\left(\sum_{v\in\gamma} m_1(v)^2\right)^{\frac{1}{2}} \cdot \left(\sum_{v\in\gamma} m_2(v)^2\right)^{\frac{1}{2}}\right]$$

Thus, by the case of equality in the Cauchy-Schwarz Inequality, there exists a $c \in \mathbb{R}$ such that $m_1(v) = c \cdot m_2(v)$. So,

$$\sum_{v\in\gamma} m_1(v)^2 = \sum_{v\in\gamma} m_2(v)^2 = \sum_{v\in\gamma}(cm_1(v))^2 = \sum_{v\in\gamma} c^2 m_1(v)^2$$

So, $c^2=1$, so $c=1$. Thus, $m_1(v) = m_2(v)$.▌

Extremal Metric Yields Square Tiling

Before delving into the theorem and proof, we must first define some things. Given a metric m on a triangulation of a region, let x(*v*) =

$$inf\left(\sum_{u\in\alpha} \mathrm{m(u)}\ :\ \alpha \text{ is a path from } \mathrm{B_2} \text{ to v}\right)$$

Likewise, let y(*v*) =

$$inf\left(\sum_{u\in\beta} \mathrm{m(u)}\ :\ \beta \text{ is a path from } \mathrm{B_1} \text{ to v}\right)$$

In other words, x(*v*) is the smallest sum of all m-values from a vertex *u* to vertex *v*. Now, we can introduce the theorem.

Theorem 3: Let T=(V,E,F;B_1,B_2,B_3,B_4) be a triangulation of a quadrilateral. Let m:V→[0,∞) be a function such that ∑m(*v*)² over all vertices is equal to inf{∑m(*v*)²}=M. Let R=[0,1] x [0,M], and for each *v*, let Z_v = [x(*v*) - m(*v*), x(*v*)] x [y(*v*) - m(*v*), y(*v*)]. Then, Z = (Z_v: *v*∈V) is a square tiling of a rectangle R which satisfies requirements (1) and (2) from **Schramm's Theorem**. Moreover, M and Z are uniquely determined by T. [**2**]

From the above theorem, we can note that each Z_v is a square of length m(*v*). Notice then that the greater value of m(*v*), the larger the square.

Proof of Theorem 3:

Let⟨*u,v*⟩be an edge in T. Since a path from B_1 to *v* can be made by attaching ⟨*u,v*⟩to any path from B_1 to *u*, we know that y(*u*) + m(*v*) ≥ y(*v*) , i.e., y(*v*) - m(*v*) ≤ y(*u*). Similarly, since a path from B_1 to *u* can be made by attaching ⟨*u,v*⟩to any path from B_1 to *v,* we know that y(v) + m(*u*) ≥ y(*u*), i.e., y(*u*) - m(*u*) ≤ y(*v*). So, [y(*u*) - m(*u*), y(*u*)] and [y(*v*) - m(*v*), y(*v*)] must overlap. Similarly, [x(*u*) - x(*u*), x(*u*)] and [x(*v*) - x(*v*), x(*v*)] must overlap by the same argument. Thus, $Z_v \cap Z_u \neq \varnothing$, where Z_v = [x(*v*) - m(*v*), x(*v*)] x [y(*v*) - m(*v*), y(*v*)] and Z_u = [x(*u*) - m(*u*), x(*u*)] x [y(*u*) - m(*u*), y(*u*)]; their sides overlap based on the argument above. This satisfies (1) of Schramm's Theorem (Theorem 1.3).

Now, set $\hat{R}_1$ = {(x,0): x ≥ 0}, $\hat{R}_2$ = {(0,y): y ≥ 0}, $\hat{R}_3$ = {(x,y): x ≥ 0, y ≥ M}, $\hat{R}_4$ = {(x,y): x ≥1, y ≥ 0}. Consider Z_v with *v*∈B_1. Then, Z_v = [x(*v*) - m(*v*), x(*v*)] x [y(*v*) - m(*v*), y(*v*)] = [x(*v*) - m(*v*), x(*v*)] x [0, y(*v*)]. Thus, $Z_v \cap \hat{R}_1 \neq \varnothing$ when *v*∈B_1. Similarly, $Z_v \cap \hat{R}_2 \neq \varnothing$ when *v*∈B_2. Now, consider *v*∈B_4. By our choice of an admissible m, we know that x(*v*) ≥ 1 for any *v*∈B_4. Thus, $Z_v \cap \hat{R}_4 \neq \varnothing$ when *v*∈B_4. We now need to show that $Z_v \cap \hat{R}_3 \neq \varnothing$ when *v*∈B_3, i.e., that if *v*∈B_3, then y(*v*) ≥ M. This will take a little bit more work.

Start by taking any curve, γ, from B_1 to B_3. Consider now , where

$$\hat{m}(w) = \frac{1}{1+t} \cdot \begin{cases} m(w) & w \notin \gamma \\ m(w) + t & w \in \gamma \end{cases}$$

and $t > 0$. Consider also any path, $\boldsymbol{\alpha}$, from B_2 to B_4. Then,

$$\sum_{w \in \boldsymbol{\alpha}} \hat{m}(w) \geq 1$$

So, $\widehat{m}(w)$ is admissible. This is due to the following reasons. We know that any path from B_2 to B_4 must cross γ. Consider

$$\widetilde{m}(w) = \begin{cases} m(w) & w \notin \gamma \\ m(w) + t & w \in \gamma \end{cases}$$

If we calculate the sum of all $\tilde{m}(w)$ on $\boldsymbol{\alpha}$, we actually calculate $[\sum m(w)] + t \geq 1 + t$ because m is admissible. So, if we divide both sides by $(1 + t)$, we find the following inequality:

$$1 \leq \frac{(\sum_{w\in\alpha} \widetilde{m}(w))}{1+t} = \frac{1}{1+t} \sum_{w\in\alpha} \begin{cases} m(w) & w \notin \gamma \\ m(w) + t & w \in \gamma \end{cases} = \sum_{w\in\alpha} \hat{m}(w)$$

This confirms that $\widehat{m}(w)$ is admissible. Now, we can calculate the following:

$$\sum_{\text{all } w \in \mathrm{V}} \hat{m}(w)^2 = \left(\frac{1}{1+t}\right)^2 \cdot \left[\sum_{w \notin \gamma} m(w)^2 + \sum_{w \in \gamma} (m(w) + t)^2\right]$$

$$= \left(\frac{1}{1+t}\right)^2 \cdot \left[\sum_{w \notin \gamma} m(w)^2 + \sum_{w \in \gamma} \left(m(w)^2 + 2m(w) \cdot t + t^2\right)\right]$$

$$= \left(\frac{1}{1+t}\right)^2 \cdot \left[\sum_{\text{all } w \in \mathrm{V}} m(w)^2 + 2t \sum_{w \in \gamma} m(w) + \sum_{w \in \gamma} t^2\right]$$

$$= \left(\frac{1}{1+t}\right)^2 \cdot \left[M + 2t \sum_{w \in \gamma} m(w) + \sum_{w \in \gamma} t^2\right]$$

Set the above equal to $g(t)$, which is a rational function, and therefore differentiable. We can now see that $g(0) = M$. Also, $g(t) \geq M$ for all $t \geq 0$ because $\widehat{m}(w)$ is admissible and m is the extremal metric. So,

$$M = \inf\left\{\sum m(w)^2\right\} \leq \sum \hat{m}(w)^2 = g(t)$$

Also,

$$g'(0) = \lim_{x\to 0} \frac{g(t) - g(0)}{t - 0} \geq 0$$

because $g(t) \geq g(0)$ for all $t \geq 0$. Now, let's examine $g'(t)$:

$$g'(t) = -2(1+t)^{-3} \cdot \left[M + 2t\sum_{w\in\gamma} m(w)^2 + \sum_{w\in\gamma} t^2\right] + \left(\frac{1}{1+t}\right)^2 \cdot \left[2\sum_{w\in\gamma} m(w) + 2t\sum_{w\in\gamma} 1\right]$$

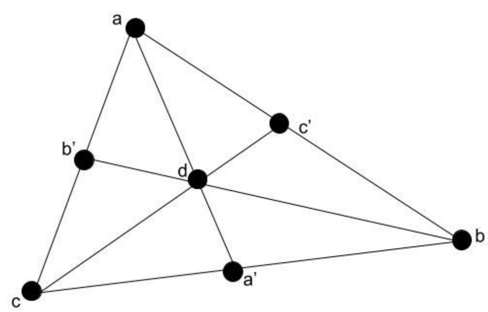
$$\Rightarrow 0 \le g'(0) = -2[M] + \left[2\sum_{w\in\gamma} m(w)\right]$$

$$\Rightarrow 0 \le -M + \sum_{w\in\gamma} m(w)$$

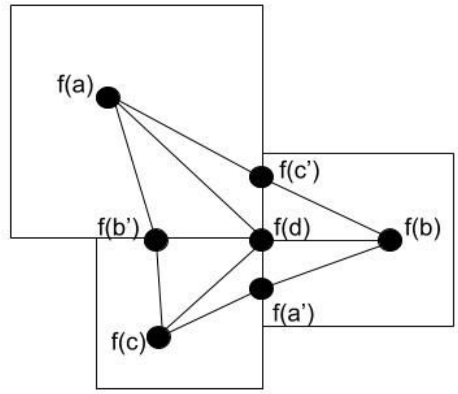
$$\Rightarrow -M \le \sum_{w\in\gamma} m(w)$$

for any γ from B_1 to B_3. So, y(*v*) ≥ M and $Z_v \cap \hat{R}_3 \neq \varnothing$.

Now, we must define a continuous function from the triangulation to the coordinate plane. Let $f: T \to \mathbb{R}^2$ be defined as follows. First, for each vertex *v*, pick a point f(*v*) in Z_v. If *v* is on the boundary of T, then f(*v*) must lie on the corresponding $\hat{R}$. Let T* be the first barycentric subdivision of T. In other words, to create T*, first place vertices at the midpoints of every edge in T. Then, connect the midpoints with the opposite vertices, creating yet another vertex in the center of each triangle where these new edges meet. Below is an example of barycentric subdivision.



Here is how we extend f to be defined on the barycentric subdivision: Consider a triangle ⟨*a,b,c*⟩. For each edge ⟨*a,b*⟩∈E, let *c'* be the midpoint of the edge. Choose a point f(*c'*) to be some point in the intersection of $Z_a \cap Z_b$. Also, it is required that $f(c') \in \hat{R}_j$ if $a,b \in B_j$. For each triangular face ⟨*a,b,c*⟩ in T, let *d* be the center of ⟨*a,b,c*⟩, and choose f(*d*) to be some point in the intersection of $Z_a \cap Z_b \cap Z_c$. Thus, the triangular faces of T* have the form ⟨*a,c',d*⟩. The union of these triangular faces form a topological disk, call it D. Now, extend the map f to T by requiring it to be affine, or linear, on each triangular face of T*. What we now have is a continuous function mapping each triangular face in T* to the “inside” of its corresponding square, as seen below:



It is clear that the image of the topological disk is contained within the union of squares, i.e.,

$$f(D) \subset \bigcup_{v \in V} Z_v \quad .$$

Now, we want to show that R=[0,1] x [0,M] is a subset of f(D). This will be done using a "winding number" (**3**). Let γ(t) be a curve that traces the boundary of T. Consider a point p∈R, and f(γ(t)) is a closed curve that does not contain p. Let $\boldsymbol{\alpha}$(t) be a continuous choice of angle between $\vec{pf(\gamma(t))}$ and the x-axis. The winding number, w, is calculated using the following equation:

$$w = \frac{\alpha(end) - \alpha(start)}{2\pi}$$

If the winding number of f(γ(t)) around p is not 0, then p is in f(D). In our case, p is contained within the rectangle R, and f(γ(t)) is just the boundary formed by the outer boundary of squares in f(T). This boundary is continuous and lies on $\hat{R}_1$ and $\hat{R}_2$, and either lies on or beyond $\hat{R}_3$ and $\hat{R}_4$. However, the boundary never ventures inside the rectangle. Let a, b, c, d be the vertices on the corners of B_1 and B_2, B_1 and B_4, B_3 and B_4, B_2 and B_3, respectively. Let t_0 occur at f(a), and let $\boldsymbol{\alpha}(t_0)$ be 0. Likewise, let t_1 occur at f(b), t_2 occur at f(c), t_3 occur at f(d), and t_4 occur at f(a). Consider now $\boldsymbol{\alpha}(t_1) - \boldsymbol{\alpha}(t_0)$, which is the measurement of the angle formed by a vertex p and the bottom corners of R. Let $\boldsymbol{\alpha}(t_1) - \boldsymbol{\alpha}(t_0) = \theta_1$. This takes care of the first segment of f(γ(t)) which lies on $\hat{R}_1$. Consider now $\boldsymbol{\alpha}(t_2) - \boldsymbol{\alpha}(t_1) = \theta_2$, which takes care of the second segment of f(γ(t)) which lies on or beyond $\hat{R}_4$. Consider $\boldsymbol{\alpha}(t_3) - \boldsymbol{\alpha}(t_2) = \theta_3$ and $\boldsymbol{\alpha}(t_4) - \boldsymbol{\alpha}(t_3) = \theta_4$, which take care of the third and fourth segments of f(γ(t)), respectively. We can now calculate w.

$$w = \frac{\alpha(t_4) - \alpha(t_0)}{2\pi}$$

$$= \frac{\alpha(t_4) - \alpha(t_3) + \alpha(t_3) - \alpha(t_2) + \alpha(t_2) - \alpha(t_1) + \alpha(t_1) - \alpha(t_0)}{2\pi} = \frac{\theta_4 + \theta_3 + \theta_2 + \theta_1}{2\pi}$$

$$= \frac{2\pi}{2\pi} = 1$$

This makes sense. We know p is within R, and f(γ(t)) follows the boundaries $\hat{R}_1$ and $\hat{R}_2$. If any loops occur on f(γ(t)), they occur beyond $\hat{R}_3$ and $\hat{R}_4$. Thus, because p is within R, the continuous choice of angle comes out to be exactly 2π. In other words, f(γ(t)) goes "around" p exactly once, so w=1. Because w≠0, then p∈f(D) for any p chosen in R. Thus, R must be contained in f(D), and is therefore a subset of f(T).

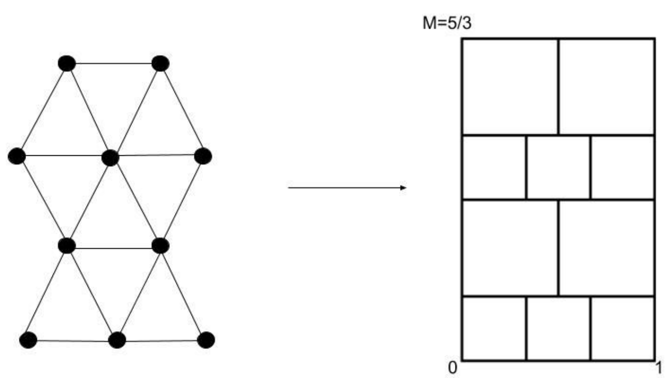
Thus, because R is a subset of f(T), and f(T) is a subset of the union of all squares, we can say that R is a subset of the union of all squares. The rectangle R has area M (calculated by multiplying side lengths of 1 and M). Each individual square Z_v has side length $m(v)^2$, so the union of all squares is calculated by the following:

$$\bigcup_{v \in V} Z_v = \sum_{v \in V} m(v)^2$$

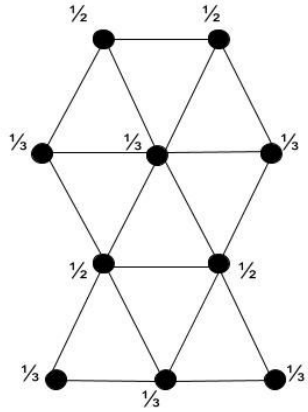
However, this is precisely the definition of modulus. Thus, the union of all squares has area equal to the modulus M, which is also equal to the area of R. Therefore, because R is a subset of the union of all squares and the two have the same area, then the union of all squares must fit perfectly inside the rectangle with no overlapping boundaries. The union of all squares is now understood to be a tiling, Z.∎

Square Tiling Yields Extremal Metric

The reverse also holds true. In other words, any square tiling corresponding to T yields an extremal metric for M(T, B_2, B_4). Given a triangulation and its square tiling, we can find the extremal metric, i.e., the assignment of m-values, that yields M. Consider the following example:



It is clear from the dimensions of the rectangle and the squares that the corresponding assignment of extremal m-values is:



The following lemma, which serves as our analog to the uniqueness statement in **Schramm's Theorem**, supports our findings:

Lemma 4: Let T=(V,E,F;B_1,B_2,B_3,B_4) be a triangulation of a quadrilateral, and let R=[0,1] x [0,M]. Let Z = (Z_v: *v*∈V) be a square tiling for T and a rectangle R, and for each *v*, let Z_v have side length m(*v*) for some m:V→[0,∞). Then m is an extremal metric for M(T, B_2, B_4).

Proof of Lemma 4:

Let m(*v*) be the side length of some square Z_v contained in the tiling Z. Consider some other admissible metric, $\dot{m}(v)$. We want to show that $\sum\dot{m}(v)^2 \geq \sum m(v)^2 = M$. Let *t* be a value between 0 and M. Let γ_t be the corresponding path in T such that all squares hit by [0,1] x {*t*} are vertices in γ_t. We know that, over all $v \in \gamma_t$, $\sum\dot{m}(v) \geq 1$. Moreover, $\sum\dot{m}(v)$ is a function of *t*. Integrating the function yields the following string of inequalities:

$$M \leq \int_0^M \left(\sum_{v\in\gamma_t} \dot{m}(v)\right) dt = \sum_{\text{all } v} \int_s^{s+m(v)} \dot{m}(v)dt$$

Here, *s* is the height of the bottom edge of Z_v. However, it doesn't matter what *s* is exactly; it is taken care of in the next step. Note that in this inequality, $\dot{m}(v)$ is not a function of *t*. This allows us to integrate fully:

$$= \sum_{\text{all } v} \dot{m}(v) \cdot m(v) \leq \left(\sum_{\text{all } v} \dot{m}(v)^2\right)^{\frac{1}{2}} \cdot \left(\sum_{\text{all } v} m(v)^2\right)^{\frac{1}{2}} \text{ (by the Cauchy-Schwarz Inequality)}$$

$$= \left(\sum_{\text{all } v} \dot{m}(v)^2\right)^{\frac{1}{2}} \cdot \sqrt{M}$$

So, $M \leq (\sum\dot{m}(v)^2)^{1/2} \cdot \sqrt{M}$. Thus, $\sum\dot{m}(v)^2 \geq M = \sum m(v)^2$. Also, since Z must come from an extremal metric on T for the path family from B_2 to B_4, and this extremal metric is unique by Lemma 2, Z is uniquely determined by T. This completes the "uniqueness" part of the proof of Theorem 3.∎

Algorithm

In [**2**], Schramm provides a 5-step algorithm that can be used to determine the unique square tiling. That algorithm is shown here:

7.1 Algorithm:

(1) Let γ_0 be a path connecting B_2 and B_4 with the fewest possible number of vertices, and let n_0 be the number of vertices in γ_0. Set $m(v) = 1/\sqrt{n_0}$ for vertices v in γ_0 and $m(v) = 0$, otherwise.

(2) Find a path γ of least m-length from B_2 to B_4, let $w_m = l_m(\gamma)$, and let n be the number of vertices in γ.

(3) If $w_m l_m = \|m\|^2$, then stop.

(4) Let $m^*(v) = m(v) + \delta$ for vertices v in γ and $m^*(v) = m(v)$ otherwise, where

$$\delta = \frac{\|m\|^2 - l_m w_m}{n l_m - w_m}.$$

(5) Replace m by $m^*/\|m^*\|$ and go to step (2). ∎

As one can see, this algorithm uses extremal length rather than modulus. In addition, many tilings require a lot (sometimes thousands) of iterations. Thus, instead of reworking this algorithm into a new, hand-written algorithm to be used with modulus, I have created a code in Mathematica that runs the algorithm as many times as can be desired. That code is listed below. Additionally, a link to the Mathematica notebook is included on the Reference Page.

Insert the specifications of your graph below, indicating which vertices share edges. Define 4 boundaries, and specify the vertices that lie on them. Then determine the number of loops you'd like to run. When finished, click "Evaluation," then "Evaluate Notebook." The output will show a labelled graph, a list of vertices, a list of corresponding colors, and a colored square tiling. The list of vertices directly corresponds with the list of colors. If vertices/colors are listed that are not seen, they have degenerated to points.

```
Original = Graph[];
B1 = {};
B2 = {};
B3 = {};
B4 = {};
NumberOfLoops = ;
```

The first section (above) is what the user sees; the user is responsible for inputting a graph (the triangulation) as well as listing the vertices on each boundary. Finally, the user determines how many loops the code should run for. I found that, usually, 500-1000 loops is enough for relatively simple graphs, but for more advanced graphs, I used 5000.

```
Graph[EdgeList[Original], VertexLabels → Table[i → i, {i, 1, Length[VertexList[Original]]}]]
phantom1 = Table[P1 → B1[[i]], {i, 1, Length[B1]}];
phantom2 = Table[P2 → B2[[i]], {i, 1, Length[B2]}];
phantom3b = Table[B3[[i]] → P3, {i, 1, Length[B3]}];
phantom4b = Table[B4[[i]] → P4, {i, 1, Length[B4]}];
OriginalEdges = EdgeList[DirectedGraph[Original]];
GEdges = Join[OriginalEdges, phantom1];
GEdges = Join[GEdges, phantom2];
GEdges = Join[GEdges, phantom3b];
GEdges = Join[GEdges, phantom4b];
G = Graph[GEdges];
vertexweights = Table[0, {i, 1, Length[VertexList[G]]}];
vertexposition[x_] := Position[VertexList[G], x][[1]][[1]];
edgeweights = Table[
   vertexweights[[vertexposition[EdgeList[G][[i]][[1]]]]],
   {i, 1, Length[EdgeList[G]]}];
WG = Graph[GEdges, EdgeWeight → edgeweights];
P = FindShortestPath[WG, P2, P4];
vertexreplacement = Table[vertexposition[P[[i]]] → 1 / Sqrt[Length[P] - 2],
   {i, 2, Length[P] - 1}];
vertexweights = ReplacePart[vertexweights, vertexreplacement];
edgeweights = Table[
   vertexweights[[vertexposition[EdgeList[G][[i]][[1]]]]],
   {i, 1, Length[EdgeList[G]]}];
```

This bit of code corresponds to Step 1 in the algorithm. Its length is due to two main reasons. Firstly, the concept of a partial boundary of vertices is one that Mathematica does not comprehend. Thus, we created phantom vertices and was careful to dictate the directions of the edges that went to them. These vertices connect to all the vertices on a given edge and have no weight; thus, instead of finding the shortest path from one boundary to another, we actually are finding the shortest path from one phantom vertec to another. I ran into issues with the "FindShortestPath" function when it began using the phantom vertices, hence the careful dictation of the directions of edges. The second issue was that Mathematica deals only with edge weights rather than vertex weights. Since we don't care about edge weights, we had to find a clever conversion between the two.

```
Do[
  WG = Graph[GEdges, EdgeWeight -> edgeweights];
  P = FindShortestPath[WG, P2, P4];
  w = GraphDistance[WG, P2, P4];
  l = GraphDistance[WG, P1, P3];
  n = Length[P] - 2;
  mass = Sum[vertexweights[[i]] ^ 2, {i, 1, Length[vertexweights]}];
  d = (mass - l * w) / (n * l - w);
  vertexreplacement =
   Table[vertexposition[P[[i]]] → vertexweights[[vertexposition[P[[i]]]]] + d,
    {i, 2, Length[P] - 1}];
  vertexweights = ReplacePart[vertexweights, vertexreplacement];
  mass = Sum[vertexweights[[i]] ^ 2, {i, 1, Length[vertexweights]}];
  vertexweights = Function[x, x / Sqrt[mass]] /@ vertexweights;
  edgeweights = Table[
    vertexweights[[vertexposition[EdgeList[G][[i]][[1]]]]],
    {i, 1, Length[EdgeList[G]]}], NumberOfLoops];
```

Above is the code that corresponds to Steps 2-5, i.e., the loop. The original algorithm has a stopping point in step 3. Instead of coding in this stopping point, which may not be reached for several thousand iterations due to miniscule errors, we run the algorithm for a certain “NumberOfLoops”, and hope the tiling comes out alright. As aforementioned, this number varies for each triangulation, but usually 1000 is plenty.

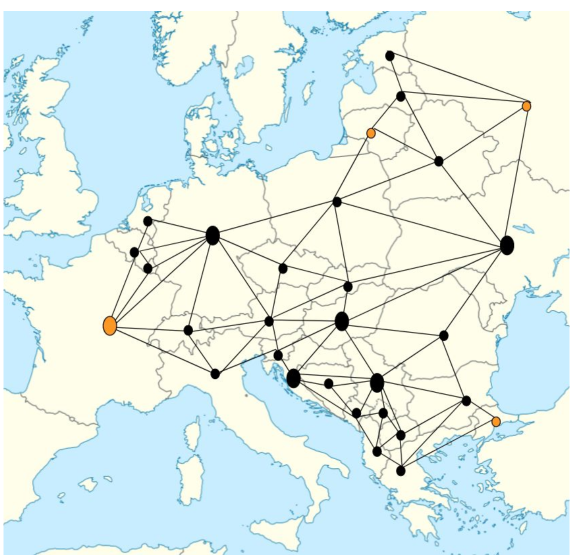
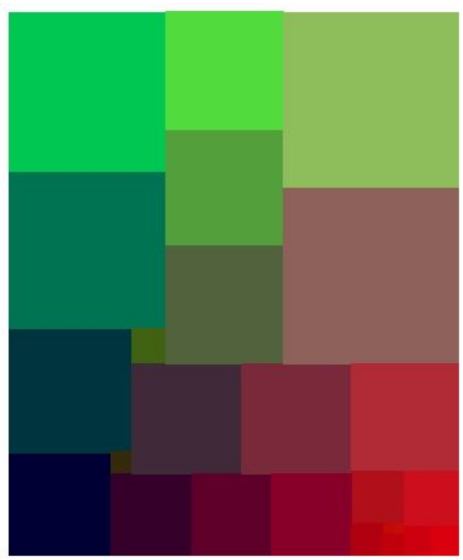
```
ColoredSquares =
  Table[{RGBColor[GraphDistance[WG, P2, VertexList[WG][[i]]],
     GraphDistance[WG, P1, VertexList[WG][[i]]], vertexweights[[i]]], Squares[[i]]},
   {i, 1, Length[Squares]}];
FinalVertexList =
  Delete[VertexList[WG], {{Length[VertexList[WG]] - 3}, {Length[VertexList[WG]] - 2},
    {Length[VertexList[WG]] - 1}, {Length[VertexList[WG]]}}];
ColoredSquaresHelpful =
  Table[{RGBColor[GraphDistance[WG, P2, VertexList[WG][[i]]],
     GraphDistance[WG, P1, VertexList[WG][[i]]], vertexweights[[i]]]},
   {i, 1, Length[Squares]}];
MapKey = Table[{FinalVertexList[[i]], ColoredSquaresHelpful[[i]]},
  {i, 1, Length[FinalVertexList]}]
Graphics[ColoredSquares]
```

This final portion of the code is not a part of the algorithm. The algorithm only finds the extremal metric; it doesn't provide instructions for drawing the square tiling. Here is the code that creates the square tiling. There are several commands here that need explaining. "ColoredSquares" yields a list of squares and their positions. This is used later in the "Graphics" command to actually create the tiling. "ColoredSquaresHelpful" is a list of just the squares without the positions; this simply looks cleaner. "MapKey" creates a list that pairs vertices from "FinalVertexList" with "ColoredSquaresHelpful". "FinalVertexList" is simply a list of vertices that doesn't include the phantoms. Also, notice that the color of each square is determined by the position of the square. This creates a gradient effect.

Examples

1. Europe

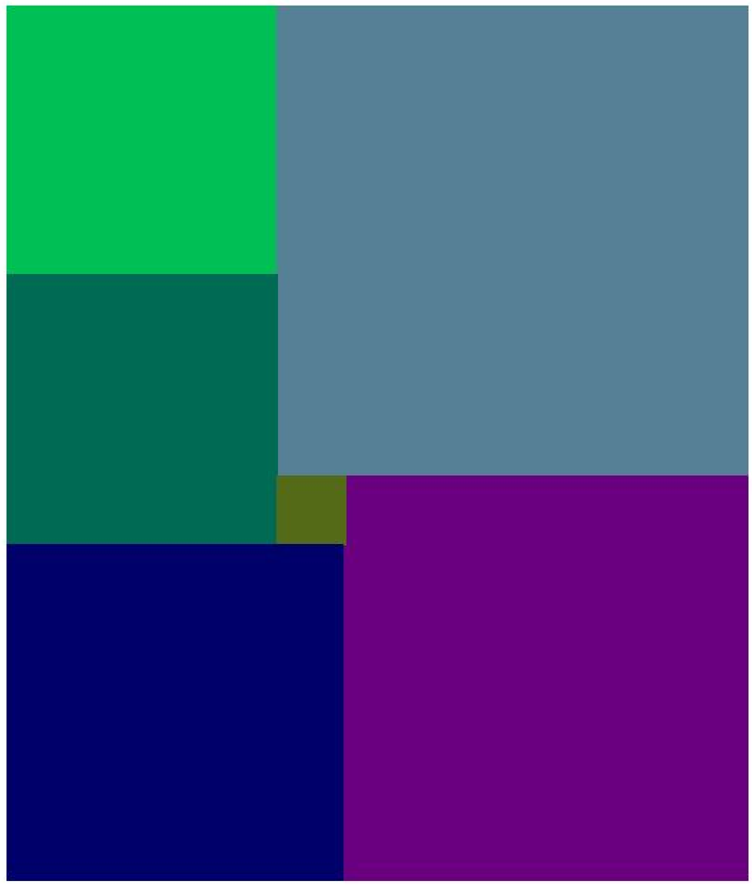
Below is a triangulation and a tiling of Europe. The boundaries are relatively close to the boundaries of the continent. B1, from right to left, runs from Turkey to France. B2, from bottom to top, runs from France to Lithuania. B3 runs from Lithuania to Russia, and B4 completes the circuit. This tiling does not include several countries in Europe: any island nations, Scandinavia, Spain, Portugal, or any nation that is completely surrounded by one or two other nations has been excluded. In each case, the country prohibits the graph from being a triangulation. A number of countries also degenerated to points.

2. South America

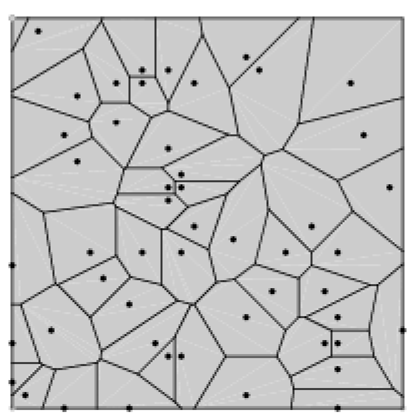
Below is a tiling of South America. The boundaries are relatively close to the boundaries of the continent. B1 is Argentina and Chile. B2 runs from Chile to Colombia. B3 runs from Columbia to Brazil, and B4 completes the circuit. Every country was included in the triangulation, but a number of countries degenerated to points; the only countries shown are Argentina, Chile, Peru, Columbia, Brazil, and Bolivia. Brazil is to blame for these degenerations.

The extremal metric is dependent on the shortest path from B2 to B4. Because Brazil is so large and has so many borders, it is almost always the destination on B4. Thus, Brazil is seen as very large in the resulting tiling.



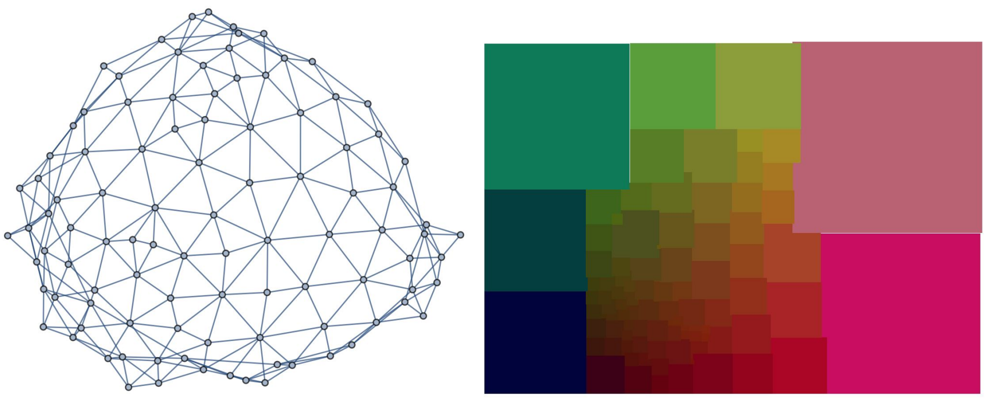
3. Delaunay Triangulations

Before describing the work done with Delaunay Triangulations, a bit of context should be provided. First, let us discuss **Voronoi Diagrams**. Given any finite set S of points in the plane, the Voronoi cells are the sets of points in the plane whose closest point in the set S is fixed [**4**]. Below is a picture of a Voronoi Diagram taken from [**4**]:

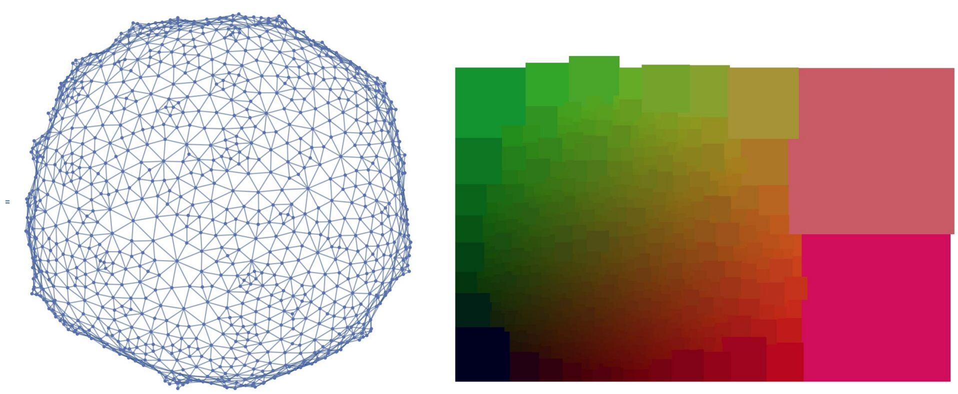


The contacts graph of the cells is known as a **Delaunay Triangulation** [**5**]. I developed a different code that generates Delaunay Triangulations based on a number of randomly generated points in a disk. I can only run the program about 100 times; for triangulations of 1000 vertices, the runtime can exceed 20 minutes. Thus, there are small gaps in the remaining tiling, but in reality, the error is small.

This code was inherently different, for more randomness had to be taken into account. This includes not only the graph itself, but also the number of vertices on the boundary. In an effort to standardize the outputs, I created the code in a way that ensured that B_4 always consisted of two vertices. Other corners are still picked randomly from a list of boundary vertices. This has an interesting effect on the resulting square tilings, as seen below:



Delaunay Triangulation with 100 vertices



Delaunay Triangulation with 1000 vertices

In line with observations, I hypothesize that as the number of vertices generated increases, the modulus decreases. This is due to the way I've coded the algorithm. Because B4 is always two vertices, as more and more vertices are used, the family of paths from B2 to B4 is squished, in a way. More vertices are forced to become smaller and smaller to accommodate the narrow paths. This holds up in running the algorithm. With 10 vertices, the modulus is around .94. With 100 vertices, the modulus is around .80. With 1000 vertices, the modulus drops further to .63.

Conformal Mappings

Another potential application of square tilings is its use as an approximation of the conformal mapping. A **conformal map** is a transformation that preserves local angles. These conformal maps have been used approximated by other packings, such as circle packings. As discussed in [**6**], converging circle packings have been used to approximate the Riemann mapping, which is a conformal map. This Riemann mapping uses something called a "hexagonal lattice," which is a way to partition a topological disk with triangles whose contacts form hexagons. Schramm mentions briefly in [**2**] that square tilings of the hexagonal mesh do not provide an approximation of the conformal mapping, but it was not determined whether there were other mappings that square tilings can approximate. One such mapping, the Schwarz-Christoffel mapping, seemed a decent candidate to explore due to its design. This mapping maps domains from the complex plane to simple polygons. Thus, the Schwarz-Christoffel mapping from the unit disk to a square seemed a reasonable candidate for attempted approximation by a square tiling.

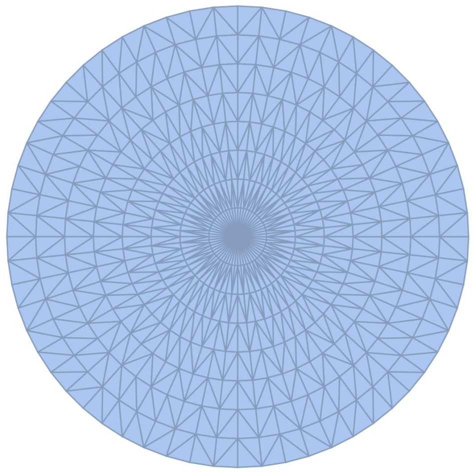
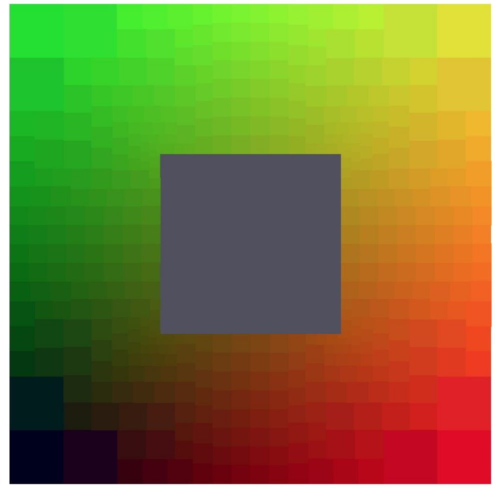
I explored the relationship between the two through another Mathematica program. I generated a circle with several radii and concentric circles, then sent the map through both a conformal function as well as through my own code. The function for a Schwarz-Christoffel mapping from the unit disk to a square, as described in [**7**], is given by:

$$F(z) = \int_0^z (1 - w^4)^{\left(-\frac{1}{2}\right)} dw$$

where *z* is a complex number in the unit disk. However, this function sends complex points to a square oriented with a corner on each axis and its center on the origin. In order to line it up with the square tiling, i.e. scale it to 1-by-1 and orient it with its bottom-left corner at the origin, the following equation was used:

$$G(z) = \left(\frac{1}{2} + \frac{i}{2}\right) + \left(\frac{1}{2} + \frac{i}{2}\right) \cdot \frac{F(z)}{F(1)}$$

I then took the average distance between the points in the conformal mapping and their corresponding squares in the tilings (using the top-right corner point of each square), and discovered that over 161 vertices, the average distance was about .14 units. This is concerning due to the fact that the corresponding square is only 1x1. Thus, I tried to do the same calculation with 481 vertices, hypothesizing that more vertices would yield smaller squares in the square tiling and a smaller average distance. This yielded better results: the average distance was about .07 units. A picture of the triangulation of the unit disk as well as the resulting tiling is shown below:

As one can see, the center square is immense in comparison to the other squares. This is a problem when attempting to approximate a conformal mapping, and the size of this square is most certainly contributing much to the average distance being .07 units. However, the size of the square is probably due to the number of concentric circles and radii used to create a triangulation of the unit circle. In this particular run, 60 radii were generated, but only 8 concentric circles. With more concentric circles, the amount of traffic running through the origin will likely decrease, resulting in smaller squares in the center of the tiling. While the question still remains open, I hypothesize that with more concentric circles and radii, the squares in the tiling will converge to points and approximate a conformal mapping.

References

[**1**] Kenneth Stephenson, *Circle Packing: A Mathematical Tale*, Notices of the AMS, **50** (2003), 1376-1388.
[**2**] Oded Schramm, *Square Tilings with Prescribed Combinatorics*, Israel Journal of Mathematics, **84** (1993), 97-118.
[**3**] Alexander F. Ritter, *Lectures on Geometry and Topology,* Oxford Masterclasses in Geometry (2014), 27-31.
[**4**] Eric W. Weisstein, *Voronoi Diagram*, WolframMathworld, http://mathworld.wolfram.com/VoronoiDiagram.html.
[**5**] Eric W. Weisstein, *Delaunay Triangulation*, WolframMathworld, http://mathworld.wolfram.com/DelaunayTriangulation.html.
[**6**] Burt Rodin and Dennis Sullivan, *The convergence of circle packings to the Riemann mapping,* Journal of Differential Geometry **26** (1987), 349-360.
[**7**] Henrik Schumacher, *Schwarz-Christoffel Maps from Unit Disk to Regular Polygons Visualization*, Mathematica Stack Exchange (2017), https://mathematica.stackexchange.com/questions/159067/schwarz-christoffel-maps-from-unit-disk-to-regular-polygons-visualization.

Digital Supplements

1) Here is a link to the code I wrote and used to generate square tilings: https://www.wolframcloud.com/obj/ajreel2/Published/Triangulations%20to%20Square%20Tilings.nb