Volume xx (200y), Number z, pp. 1–12

Graph Embedding by Square Tiling

Jialing Zhang [1], Wei Chen [1], Junwei Zhang[2] and Xianfeng Gu[2]

[1]Kunming University of Science and Technology, China
[2]Stonybrook University, Department of Computer Science, USA

Abstract

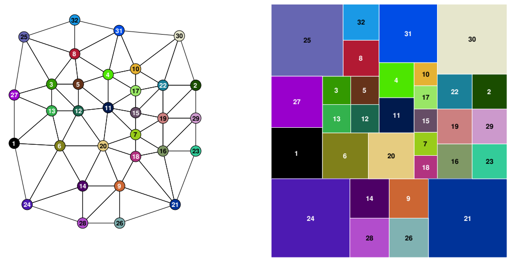
A novel and general planar graph embedding method is proposed, which is based on square tiling by quadratic programming with convex constraints. The method maps a given planar graph to a square tessellation of a rectangle, such that each node is represented by a square, two adjacent nodes are mapped to two squares in contact. The method is based on the extremal length theory in conformal geometry. The algorithm has two stages, first the graph is embedded using discrete Ricci flow method, which gives the initial estimation of the embedding, then the square tilling is computed by optimizing a quadratic energy with convex constraints. The embedding has many merits for graph visualization: each node is mapped to a square, it is convenient to add labels or images for the node; the edges are encoded by the cell adjacency relation, so the embedding has no edge-crossings and is more intuitive to visualize graphs; each square can be further tessellated by squares, then the nested square tiling can be applied for visualizing hierarchical graphs. As a result, We will employ multiple application scenarios to demonstrate the efficacy, efficiency and robustness of our graph embedding method.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Shape modeling—Parametric curve and surface models

1. Introduction

Abstract node-link graphs are inadequate in many visualization applications where nodes are frequently equipped with labels or icons etc. Since points only convey information on neighborhood relations, there is necessary to employ other geometric objects such as boxes or discs as visual resources to convey additional information contained in the underlying data. However, directly replacing nodes with those larger geometric objects may cause considerable overlap and poor readability in the derived two-dimensional layout. A large variety of visualization tasks, for instance, word-cloud construction, graph drawing and label placement, essentially focus on arranging planar geometric objects so as to retain similar or neighboring structures of the underlying data. Although significant advances have been made towards producing meaningful layouts from geometric objects, there is still off-balance use on neighborhood structure preserving, overlap removal, flexible scalability, and excellent readability etc. Therefore, finding an optimal balance among multiple requirements is still a challenging task in computer vision and graphics [GNCMH16].

In this paper, we propose a novel method from the viewpoint of conformal geometry to create planar layouts for original node-link graphs, which is not only able to reach a wide range of visualization requirements but also preserves the shape information of original graph. The derived layouts are made up of varying size squares with overlap removal and compact pattern. More importantly, the generic geometric context of original skeletons are preserved in this two-dimensional displays. So the problem of directly replacing nodes with other geometric objects meanwhile preserving neighboring structure in two-dimensional layouts is perfectly solved. The proposed method originates from discrete extremal length theory, which is related to quadratic optimization with convex constraints, thus the existence and uniqueness of solution can be guaranteed.



(a) Input graph (b) Square tiling

Figure 1: *Schematic representation of the concept of graph embedding by square tiling: (a)The original node-link graph with triangular mesh structure; (b)The planar representation is generated by the proposed embedding technique, which represents each node as a square and describes the link relation between data as corresponding squares touching with disjoint interiors.*

Since the proposed method is completely different from existing methods for planar graph representation, we hope that a simple explanation of its mechanism will benefit comprehension. Given an internally triangulated graph G, we manually select any four nodes on its boundary as shown in Fig.1, e.g.$\{25,30,21,24\}$, which divide the boundary into four sides, indicated as left, right, top and bottom side separately. Considering all possible paths from left side to right side, the family of these paths is denoted as Γ. We assign a positive weight ρ at each node, the so called metric, which defines a square with its edge length equal to $\rho(v_i)$ at the node v_i. After arranging those squares in plane without overlap, we obtain an alternate representation of G. Furthermore, the representation is unique based on the following mathematical principle: Let $\gamma \in \Gamma$ be a path from left side to right side, consisting of nodes $\{v_{i_1}, v_{i_2}, \cdots, v_{i_k}\}$, the length of the path under ρ is the summation of the edge lengths of relevant squares:

$$L_\rho(\gamma) = \sum_{j=1}^{k} \rho(v_{i_j}). \tag{1}$$

The total area of the derived layout is the summation of the areas of all squares,

$$A(\rho) = \sum_{v \in G} \rho^2(v). \tag{2}$$

The *extremal length* of the path family Γ is defined as

$$EL(\Gamma) := \sup_{\rho} \frac{\inf_{\gamma \in \Gamma} L_\rho^2(\gamma)}{A(\rho)}. \tag{3}$$

Intuitively, Eqn.3 is to find the maximum ratio between the square of the shortest path length and the total area. According to geometric theory, the optimal ρ that realizes the extremal length of Γ induces the square tiling representation of G. From a computational point of view, computing the extremal length is equivalent to minimizing a quadratic energy $A(\rho)$ under the linear constraints $L_\rho(\gamma) \geq 1$ for all $\gamma \in \Gamma$. Because the quadratic form is positively definite, the solution exists and is unique. In practice, we can use active set method to solve the quadratic programming problem, which converts inequality constraints to equality constraints. Estimating the active set of the constraints is crucial to improve the efficiency, which means finding the shortest path. In turn, the active set can be accurately estimated by finding a good approximation to the optimal metric ρ, which can be obtained by using discrete Ricci flow method. Consequently, the original graph G is uniquely mapped into a rectangle tiled by varying size squares wherein each square represents a node and the link relation between nodes is depicted as corresponding squares in contact with disjoint interiors. In some sense, the problem of finding a two-dimensional layout of a skeleton with no self-intersection can be considered as a planar graph embedding problem [MK16]. So we will naively call the proposed method as graph embedding by square tiling.

The main contributions of current work include:

1. To the best of our knowledge, this is the first work that represents a internally triangulated graph with four fixed boundary points as a rectangle tiled by squares under conformal transformation.
2. This is the first work that applies graph embedding by square tiling to realize versatile requirements in two-dimensional visualization layouts. And we will also apply the proposed method to represent multi level structures and construct conformal surface parameterization.
3. A practical algorithm based on the quadratic programming with convex constraints has been introduced, which can produce the square tiling representation for general triangle graphs.
4. Circle packing graph embedding method based on discrete Ricci flow is proposed to approximate the optimal metric, which significantly improves the computational efficiency.

This paper is organized as follows: section 2 briefly reviews the most related works; section 3 explains the theoretical foundation of the current work; section 4 exposes the algorithmic details of the computational pipeline; section 5 demonstrates the experimental results in different applications; section 6 gives a conclusion.

2. Related work

2.1. Contact Representation for Plane Graph

There is a large amount of work on planar node-link graph represented by alternative contact graph, where vertices are represented by geometric objects, such as circles [Koe36], triangles [KMN12] and rectangles [KK88, LL84], and edges correspond to two objects touching in some specified fashion. Rectangle representations is one of the most common representations for practical applications, cartography [Rai34], geography [Tob04], sociology [HK98]and floor-planning for VLSI layout [YS93] etc. Generally, rectangle representations for unweighted graphs are viewed as rectangular duals which dissect a rectangle into smaller rectangles or squares, while rectangle cartograms for weighted graphs design edge or vertex weights propotional to edge lengths or areas of rectangles without changing combinatorial structures of graphs. A historical overview and a summary of the state of the art in the rectangle contact graphs literature can be found in Buchsbaum et al. [BGPV05] and Felsner [Fel12].

Unlike the previous work, we will provide a Square tiling representation for an internally triangulated planar graph in the viewpoint of conformal geometry, where shape information of a triangular mesh can be preserved. In mathematics literature, the theory related to square tiling view for a planar triangle mesh can be traced back to a seminal paper *The dissection of rectangles into squares* by Brooks, Smith, Stone and Tutte in [Bro40]. They used a physical model of current flows to show whether a square can be tiled by smaller squares whose edge-lengths are all different. This appealing problem produced a beautiful connection between square tilings and harmonic functions. Cannon, Floyd and Parry established combinatorial Riemann mapping theorem based on square tiling in [Can94, CFP94]. Schramm developed similar square tiling theory from certain kinds of extremal length problems and quadratic optimization on convex domains [Sch93]. Our work is based on the work of Schramm to produce the planar graph embedding by square tiling method.

2.2. Visualization Layouts by Geometric Objects

Tiling the visual space with geometric objects to produce a well structured layout is a common technique in visualization applications. Treemap [JS91] and its variants [FP02, BHVW00, SW01, Shn13] are examples of depicting hierarchical structures by a sequence of nested rectangles. The visualization is performed by recursively splitting the space in rectangular boxes whose areas are proportional to quantitative attributes of the data, while size and orientation reflect the extent of nodes in the hierarchy. These methods have advantage of the efficiency in space occupation, but they cannot perform well in placing similar item close to each other.

Another method is node-displacement technique which seeks to arrange geometric objects in the positions of nodes with no overlap and preserving similarity relations. According to different mechanisms, they are usually identified as physical-based, optimal-based and heuristic-based models. Physical-based models apply force schemes [GH09, HLSG10] or spring systems [HLSG10, CLY09] and iteratively minimize a whole energy function for placing geometric objects in the visual space. The weaknesses of this kind of methods are related to no guarantee of computational convergence and untidiness of the resulting layouts. Optimal-based methods minimize the quadratic energy with some constrains to solve the problem of convergence, such as methods proposed by Dwyer et al. [DMS06] and Marriott et al. [MSTH03]. Heuristic methods arrange geometric objects with known positions in post-proceeding step, for example, RWordles [SSS*12] using similarity information to build semantically aware layouts. Optimal and heuristic methods have advantages on overlap removal and similarity preservation in the derived layouts, but easliy causing the issue on efficient usage of display area. Generally speaking, node-displacement techniques lack the scalability to design layouts for hierarchical structures. Relying on simultaneously applying multidimensional projection, density-based adaptive grids, and mixed integer optimization, the latest work [GCM*16] claimed that the derived layouts can concurrently support multiple requirements, i.e. semantically aware, usage of display area, hierarchy representation and well grid-like structure.

Previous work has never considered to map the planar internally triangulated graph into a rectangle tiled by squares from the conformal viewpoint. In the current work, we will point out that such consideration have advantages on these aspects: overlap removal, compact layout, hierarchy representation, guaranteed convergence, and uniqueness of solution. Furthermore, the proposed method produces a fantastic weight for each vertex. Taking these weights as edge lengths of squares, we are able to construct a hole-free and shape-preserved combinatorial image for 3D triangle mesh, which is topologically equivalent to a quadrilateral.

3. Theoretic Foundation

3.1. Conformal Mapping

Suppose S is a topological surface, a *Riemannian metric* $\mathbf{g}$ assigns inner product on the tangential plane at each point. Assume (x_1,x_2) to be local coordinate of the surface, the metric can be represented as a matrix-valued function, $\mathbf{g}(x_1,x_2) = (g_{ij}(x_1,x_2))$, where the matrix (g_{ij}) is positive definite at every point. Suppose $\mathbf{h} = (h_{ij})$ is another metric on S. If there exists a function $\lambda : S \to \mathbb{R}$, such that

$$\mathbf{h} = e^{2\lambda}\mathbf{g},$$

we say $\mathbf{h}$ is *conformal* to $\mathbf{g}$. The function λ is called the *conformal factor*. Namely, the angle values measured by $\mathbf{h}$ equals to that by $\mathbf{g}$.

Suppose $(S,\mathbf{g})$ and $(T,\mathbf{h})$ are two Riemannian surfaces with local coordinate (x_1,x_2) and (y_1,y_2) respectively. $\varphi : S \to T$ is a diffeomorphism between them with local representation $(x_1,x_2) \mapsto (y_1,y_2)$. The *Jacobian matrix* of the map is given by $J = (\partial y_i/\partial x_j)$. The mapping φ induces a *pull back metric* on the source as

$$\varphi^*\mathbf{h} = J^T\mathbf{h}J = \left(\sum_{k,l}\frac{\partial y_i}{\partial x_k}h_{kl}\frac{\partial y_l}{\partial x_j}\right)$$

Definition 1 (Conformal Mapping) Suppose $\varphi : (S,\mathbf{g}) \to (T,\mathbf{h})$ is a smooth mapping between two Riemannian surfaces, if the pull back metric induced by φ is conformal to the original metric,

$$\varphi^*\mathbf{h} = e^{2\lambda}\mathbf{g},$$

where $\lambda : S \to \mathbb{R}$ is a function, then the mapping φ is called a conformal mapping.

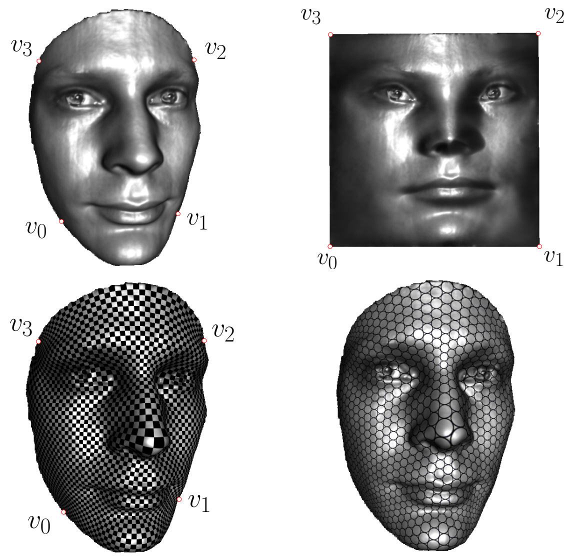


Figure 2: *Schematic representation of conformal transformation for a face surface.*

Fig.2 shows one example of conformal mapping from a human facial surface to a planar rectangle. We put a checker-board texture on the rectangle, and the mapping pulls the texture back onto the facial surface. It is obvious that all the right corner angles of the checkers are well-preserved. We put circle packing texture on the rectangle, the circular shapes are also well preserved. This shows a unique property of conformal mapping: it maps infinitesimal circles to infinitesimal circles.

For each point p on the surface S, we can find a neighborhood $U(p)$, such that the neighborhood can be conformally mapped onto the planar disk, which induces a special local coordinates (x,y) on $U(p)$, the so-called *isothermal coordinates*. The metric can be written as

$$\mathbf{g} = e^{2\lambda}(dx^2 + dy^2)$$

The *Gaussian curvature* of the surface is given by

$$K(x,y) = -e^{-2\lambda(x,y)}\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)\lambda(x,y)$$

The total Gaussian curvature satisfies the Gauss-Bonnet theorem,

$$\int_S K(x,y)e^{2\lambda}dxdy = 2\pi\chi(S)$$

where $\chi(S)$ is the Euler-characteristic number of the surface.

There are different approaches to obtain a conformal mapping such as Ricci flow. Here we will take a novel method: extremal length and introduce its theory and algorithm.

3.2. Ricci Flow

Ricci flow conformally deforms the surface metric proportional to the curvature, such that the curvature evolves according to a non-linear heat diffusion process, and eventually becomes constant everywhere. The analytic formulation of Ricci flow is given by

$$\frac{dg_{ij}(p,t)}{dt} = 2\left(\bar{K}(p)-K(p,t)\right)g_{ij}(p,t)$$

where $\bar{K}(p)$ is the target curvature at the point p. Because the deformation is conformal $\mathbf{g}(p,t) = e^{2\lambda(p,t)}\mathbf{g}(p,0)$, the Ricci flow is

$$\frac{d\lambda(p,t)}{dt} = \bar{K}(p)-K(p,t). \quad (4)$$

Hamilton and Chow proved the convergence of surface Ricci flow.

Theorem 1 (Hamilton and Chow) Suppose $(S,\mathbf{g})$ be a closed compact surface, the normalized surface Ricci flow with target curvature $\bar{K}(p) = 2\pi\chi(S)/A(0)$, where $\chi(S)$ is the Euler characteristic number of S, $A(0)$ is the initial total area of the surface, converges.

Ricci flow is a powerful tool to design a Riemannian metric by prescribing the curvature.

3.2.1. Discrete Ricci Flow

Discrete Ricci flow generalizes surface Ricci flow to the discrete setting. Suppose $M = (V,E,F)$ is a polyhedral surface, each face is a Euclidean triangle, V,E,F represent the set of vertices, edges and faces respectively. A *discrete Riemannian metric* of M is represented as the edge length function $l : E \to \mathbb{R}^+$, which satisfies the triangle inequality on each face. The *discrete Gaussian curvature* is defined as the angle deficit on each vertex,

$$K(v_i) = \left\{\begin{array}{ll} 2\pi - \sum_{jk}\theta_i^{jk} & v_i \notin \partial M \\ \pi - \sum_{jk}\theta_i^{jk} & v_i \in \partial M \end{array}\right.$$

where θ_i^{jk} is the corner angle at v_i in triangle $[v_i,v_j,v_k]$. The total curvature satisfies the Gauss-Bonnet theorem

$$\sum_{v_i\in V} K(v_i) = 2\pi\chi(M).$$

We associate each vertex v_i with a disk centered at v_i with radius r_i and define the edge length as

$$l([v_i,v_j]) = r_i + r_j,$$

which is called the *tangential circle packing metric*. The *discrete Ricci flow* is given by

$$\frac{dr_i}{dt} = \left(\bar{K}(v_i)-K(v_i)\right)r_i,$$

In fact, this is the gradient flow of a convex energy, the so-called entropy energy. Let $u_i = \log r_i$, $\mathbf{u} = (u_1,u_2,\cdots,u_n)$, the entropy energy is given by

$$E(\mathbf{u}) = \int_{\mathbf{u_0}}^{\mathbf{u}}\sum_{i=1}^{n}(\bar{K}(v_i)-K(v_i))du_i.$$

The gradient of the entropy energy is

$$\nabla E(\mathbf{u}) = (\bar{K}_1-K_1,\bar{K}_2-K_2,\cdots,\bar{K}_n-K_n)^T.$$

Suppose one interior edge $[v_i,v_j]$ is adjacent to two faces $[v_i,v_j,v_k]$ and $[v_j,v_i,v_l]$, the in-circle radii of them are τ_k and τ_l respectively, then the edge weight

$$w_{ij} := \frac{\tau_k+\tau_l}{r_i+r_j}.$$

If the edge is on the boundary, and only adjacent to one face $[v_i,v_j,v_k]$, then

$$w_{ij} := \frac{\tau_k}{r_i+r_j}.$$

The Hessain matrix of the energy is given by

$$\frac{\partial^2 E(\mathbf{u})}{\partial u_i \partial u_j} = \left\{\begin{array}{rl} 0 & i\neq j, v_i \not\sim v_j \\ w_{ij} & i\neq j, v_i \sim v_j \\ -\sum_k w_{ik} & i = j \end{array}\right.$$

The energy is concave on the space $\{\mathbf{u}|\sum_i u_i = 0\}$, and can be optimized using Newton's method efficiently.

Suppose S is a topological disk (simply connected surface with a single boundary), with four corner vertices $\{v_0,v_1,v_2,v_3\} \subset \partial S$ sorted counter-clock-wisely on the boundary, we denote it as $(S,\{v_0,v_1,v_2,v_3\})$. Such kind of surface is called a *topological quadrilateral*. By applying Ricci flow, one can show the following theorem.

Theorem 2 Suppose $(S,\{v_0,v_1,v_2,v_3\})$ is a topological quadrilateral with a Riemannian metric $\mathbf{g}$, then there is a conformal mapping $\varphi : S \to \mathscr{R}$, where $\mathscr{R}$ is a planar rectangle, such that the four corner vertices are mapped to the corners of the rectangle.

Proof The four corner vertices partition the boundary into four segments $\{s_0,s_1,s_2,s_3\}$, where s_k connects the corners v_{k-1} and v_k, where $k = 1,2,3$. We make a copy of S, and reverse its orientation, denoting it as $\bar{S}$. Then we glue S and $\bar{S}$ along s_0 and s_2 to obtain a topological cylinder. This process is called a double covering of the quadrilateral. Similarly, we can double cover the cylinder to obtain a topological torus. Then by Ricci flow, the torus can be conformally deformed to a flat torus with a Euclidean metric. By symmetry, the image of S in the flat torus is a planar rectangle. □

Definition 2 Suppose the width and the height of $\mathscr{R}$ are w and h respectively, we call the *conformal module* of $(S,\{v_0,v_1,v_2,v_3\})$ to be w/h.

3.3. Extremal Length

There is another way to show that a topological quadrilateral can be conformally mapped onto a planar rectangle, the so-called *extremal length* method, which formulates the problem of finding the conformal mapping as an optimization problem.

We consider the collection of smooth paths on S connecting the left side to the right side

$$\Gamma := \{\tau : [0,1] \to S | \tau(0) \in s_0,\ \tau(1) \in s_2\}, \tag{5}$$

Definition 3 (Extremal Length) Let $(S, \mathbf{g})$ is a topological disk, Γ in Eqn. 5 is a collection of paths. Suppose ρ is another Riemannian metric conformal to $\mathbf{g}$, $\rho = e^{2\lambda}\mathbf{g}$. Let

$$L_\rho(\Gamma) := \inf_{\gamma \in \Gamma} L_\rho(\gamma),$$

where $L_\rho(\gamma)$ is the length of a path γ under the metric ρ. Let $A(\rho)$ be the area of S under the metric ρ, then the extremal length of Γ is

$$EL(\Gamma) := \sup_{\rho \sim \mathbf{g}} \frac{L_\rho^2(\Gamma)}{A(\rho)}, \tag{6}$$

where the supremum is over all Riemannian metrics conformal to $\mathbf{g}$, such that $0 < A(\rho) < \infty$.

By definition, we can see that extremal length is a conformal invariant. In fact, the extremal length equals to the aspect ratio of the rectangle. The flat Riemannian metric on the rectangle is the maximizer of the function in Eqn.6.

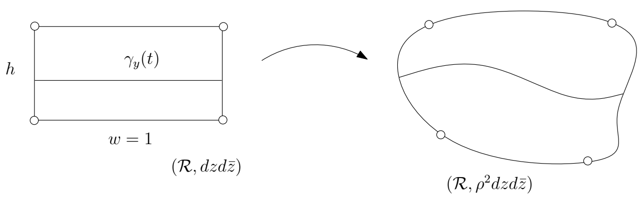


Figure 3: *Extremal length for a planar rectangle equals to w/h.*

Lemma 1 Let $\mathscr{R}$ be a planar rectangle with width and height w and h respectively. The path family Γ is given in Eqn.5, then

$$EL(\Gamma) = w/h.$$

Proof As shown in Fig.3, without loss generality, we can scale the rectangle such that the width equals to 1, and still denote the height as h. First, we may take $\rho = 1$ on $\mathscr{R}$, then $A(\rho) = h$ and $L_\rho(\Gamma) = 1$. The definition of $EL(\Gamma)$ as a supremum then gives $EL(\Gamma) \geq 1/h$. Now consider any Riemannian metric $ds^2 = \rho^2(dx^2 + dy^2)$, such that $L_\rho(\Gamma) = 1$ and $0 < A(\rho) < \infty$. Considering a horizontal line $\gamma_y(t) = \gamma(t, y)$, then

$$1 \leq \int_0^1 \rho \circ \gamma_y(x) dx = \int_0^1 \rho(x,y) dx,$$

integrating this inequality over all $y \in (0, h)$ implies

$$h \cdot 1 \leq \int_0^h \int_0^1 \rho(x,y) dxdy,$$

applying Cauchy-Schwartz inequality gives

$$h \leq \int_0^h \int_0^1 \rho(x,y) dxdy \leq \left(\int_{\mathscr{R}} \rho^2 dxdy \int_{\mathscr{R}} dxdy \right)^{1/2} = (A(\rho)h)^{1/2},$$

namely, $A \geq h$, $EL(\Gamma) = 1/A \leq 1/h$. Therefore, combining two inequalities, we obtain $EL(\Gamma) = 1/A$ as required. Furthermore, when the Cauchy-Schwartz inequality becomes an equality, ρ is constant everywhere. Namely, the conformal Riemannian metric is flat. □

In the above argument, we apply Cauchy-Schwartz inequality, which can be interpreted as follows. In Euclidean space, the inner product of two vectors $\mathbf{v}, \mathbf{w} \in \mathbb{R}^n$ is

$$\langle \mathbf{v}, \mathbf{w} \rangle = \|\mathbf{v}\| \cdot \|\mathbf{w}\| \cos\theta \leq \|\mathbf{v}\| \cdot \|\mathbf{w}\|,$$

where θ is the angle between the two vectors. When equality holds, $\theta = 0$ and two vectors are parallel to each other, there is a constant $\lambda \in \mathbb{R}$, $\mathbf{w} = \lambda\mathbf{v}$.

In the functional space $L^2(\mathscr{R})$, the inner product of two functions $f, g \in L^2(\mathscr{R})$ is given by

$$\langle f, g \rangle := \int_{\mathscr{R}} f \cdot g\, dxdy.$$

then the norm of a function is given by

$$\|f\| := \sqrt{\langle f, f \rangle}$$

Then Cauchy-Schwartz inequality means

$$\langle f, g \rangle \leq \|f\| \cdot \|g\|,$$

similarly, the equality holds $g = \lambda f$ for some constant $\lambda \in \mathbb{R}$.

3.4. Square Tiling

Square tiling can be treated as the generalization of extremal length in the discrete setting. Given a triangulated graph $G = (V, E, F)$, V, E, F are denoted as the set of nodes, edges, and faces respectively. We choose four corner nodes $\{v_0, v_1, v_2, v_3\}$ on the boundary of the graph. Then the boundary is divided into four segments $\{s_0, s_1, s_2, s_3\}$. A path γ on the graph is a consecutive sequence of vertices $\{v_{i_0}, v_{i_1}, \cdots, v_{i_k}\}$, where v_{i_l} and $v_{i_{l+1}}$ are connected by an edge in E. We denote the family of paths connecting any node in segment s_0 and any one in segment s_2 as

$$\Gamma := \{\gamma = \{v_{i_0}, v_{i_1}, \cdots, v_{i_k}\} | v_{i_0} \in s_0, v_{i_k} \in s_2\}.$$

Let $\rho : V \to [0, \infty)$ be a non-negative function defined on the node set V. It can be treated as the discrete conformal factor. Denoting $\rho(v_i)$ by ρ_i, the length of a path $\gamma = \{v_{i_0}, v_{i_1}, \cdots, v_{i_k}\}$ is defined in Eqn. 1, and the area of the whole graph under ρ is given by Eqn. 2. The discrete extremal length of $(G, \{v_0, v_1, v_2, v_3\})$ is defined in Eqn. 3. This extremal length induces a square tiling of the graph.

Definition 4 (Square Tiling) Let $G = (V, E, F)$ be a planar graph with four corner nodes $\{v_0, v_1, v_2, v_3\}$ selected on its boundary. A square tiling of $(G, \{v_0, v_1, v_2, v_3\})$ is a cell decomposition of a rectangle, such that

1. each node corresponds to a square in the rectangle,
2. if two nodes are adjacent in G, their corresponding squares are tangent to each other,
3. four corner vertices correspond to four squares at the corners of the rectangle.

Theorem 3 (Square Tiling) Let $G=(V,E,F)$ be a planar graph with four corner nodes $\{v_0,v_1,v_2,v_3\}$ selected on its boundary. There exists a square tiling of $(G,\{v_0,v_1,v_2,v_3\})$ and the square tiling is unique up to a scaling.

The proof can be found in [Sch93] and [Can94].

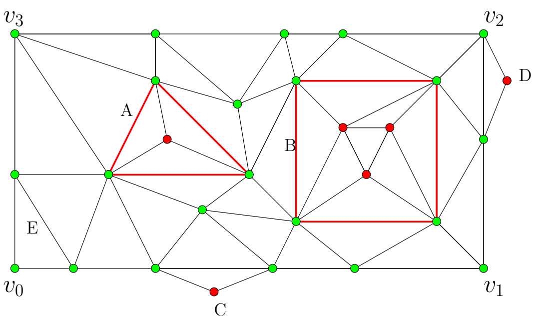


Figure 4: *The green nodes generate nontrivial squares while red nodes cause trivial squares, which are degenerate into points.*

As shown in Fig.4, the squares corresponding to some nodes may shrink to points, which are called *degenerate nodes*. The green nodes are normal ones, while the red nodes are degenerate. The figure illustrates the possibly degenerate scenarios:

1. A vertex is separated by a 3-cycle as shown in case A;
2. Vertices are separated by a 4-cycle as shown in case B;
3. All vertices of a triangle are on the boundary of the graph, and none of them is a corner vertex, then at least one of them is degenerate, as shown in case C;
4. All vertices of a triangle are on the boundary of the graph, and one is a corner vertex whilst the other two on the same side of $s_k (k=0,1,2,3)$, then the corner vertex is degenerate, as shown in case D.

The detailed proof can be found in [Sch93].

3.5. Quadratic Programming

The computation of the extremal length of a graph is equivalent to a quadratic programming problem. The objective is to minimize the area $A(\rho)=\sum_i \rho_i^2$, which is quadratic to the unknown $\rho=(\rho_0,\rho_1,\cdots,\rho_n)$, with the convex constraints: for each node v_i, $\rho_i \geq 0$; for each path $\gamma_k \in \Gamma$, the length $L_\rho(\gamma_k)\geq 1$. So

Definition 5 (Convex Quadratic Programming) The quadratic programming with convex linear constraints can be formulated as

$$\min_\rho \sum_{i=0}^n \rho_i^2$$

s.t.

$$\begin{array}{c} \rho_i \geq 0, \forall i \\ \sum_{j=0}^k \rho_{i_j} \geq 1, \forall \gamma_k \in \Gamma \end{array} \tag{7}$$

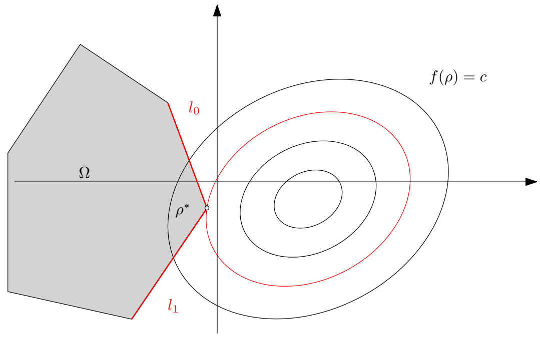


Figure 5: *The solution of quadratic optimization problem with convex constraints is the touching point ρ^* between the boundary of admissible space Ω and the level set $f(\rho)=c$.*

As shown in Figure 5, each inequality constraint defines a half space. The intersection of all half spaces is a convex polyhedral set Ω in the Euclidean space, which is called the *admissible space*. The level set of the quadratic energy function $A(\rho)=C$ is an ellipsoid, so the minimal solution is the touching point ρ^* between the boundary of the admissible space and the level set. The convexity of the admissible space guarantees the uniqueness of the solution. At the intersection point ρ^*, the faces of the admissible space form the active-set $\{l_0,l_1\}$ as shown in the Figure 5. The gradient of the convex energy is a linear combination of the normals of the hyper planes in the active-set, which is the so-called Karush-Kuhn-Tucker (KKT) condition. Convex quadratic programming can be efficiently optimized using active-set method.

The general quadratic programming problem can be formulated as:

$$\min_{\mathbf{x}} g(\mathbf{x})=\frac{1}{2}\mathbf{x}^T G\mathbf{x}+\mathbf{x}^T\mathbf{c}$$

s.t.

$$\begin{array}{rcl} \mathbf{a}_i^T\mathbf{x} & = & \mathbf{b_i} \\ \mathbf{a}_j^T\mathbf{x} & \geq & \mathbf{b_j} \end{array}$$

The Lagrangian is

$$\mathscr{L}(\mathbf{x},\lambda)=\frac{1}{2}\mathbf{x}^T G\mathbf{x}+\mathbf{x}^T\mathbf{c}-\lambda^T(A\mathbf{x}-\mathbf{b}),$$

where

$$A=\begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_m \end{bmatrix} \in \mathbb{R}^{m\times n}$$

The KKT condition can be formulated as

$$\begin{array}{rcl} \nabla\mathscr{L}(\mathbf{x},\lambda) & = & \mathbf{0} \\ \mathbf{a}_i^T\mathbf{x} & = & \mathbf{b}_i \\ \mathbf{a}_j^T\mathbf{x} & \geq & \mathbf{b}_j \\ \lambda_i & \geq & 0 \\ \lambda_i(\mathbf{a}_i\mathbf{x}-b_i) & = & 0 \end{array}$$

If G is positively defined, then there is a unique global solution of the optimization problem. If there are only equality constraints, from $\nabla\mathscr{L}(\mathbf{x},\lambda)=\mathbf{0}$, we obtain the optimal solution

$$\begin{bmatrix} G & -A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix}=\begin{bmatrix} -\mathbf{c} \\ \mathbf{b} \end{bmatrix}.$$

The active set method solves constrained optimization problems by searching solutions in the feasible sets. If constraints are linear and one can guess the active constraints for the optimal solution, then one can use the active constraints to reduce the number of unknowns, and change inequality constraints into equality constraints, further perform algorithms for unconstrained optimization problems.

In practice, it is crucial to choose the family of paths Γ, or equivalently, the active-set in the optimization. The brute-force way to consider all the paths in Γ is highly inefficient. One way to prune spurious paths is to use circle packing method. We can compute the conformal mapping using discrete Ricci flow method based on tangential circle packing. We set the target curvature to be zero for all interior vertices, and $\pi/2$ for four corner vertices, and zeros for all other boundary vertices. The tangential circle packing gives us a solution, the circle radius on v_i can be treated as a good initial approximation of ρ_i. Then we compute the shortest path from each node on the left side s_0 to the right side s_1 using the circle radii as the metric. The collection of these shortest paths are used as Γ. This greatly improves the efficiency.

4. Computational Algorithm

This section explains the algorithm pipeline in detail. The input is a triangulated graph G with four corner vertices $\{v_0,v_1,v_2,v_3\}$ on its boundary. The central task is to compute the extremal length of all the paths Γ connecting the left and right sides of G, which is to minimize a quadratic energy with linear constraints.

In our problem, the curve family Γ consists of all paths connecting the left and right sides of G, and each path γ corresponds to a linear inequality constraint. Under the optimal metric, the active set solely includes the shortest path connecting left and right sides, and the inequality constraints become equality constraints. It is crucial to determine the shortest path on the active set. But it becomes a "chicken-egg" problem: in order to decide the active set, we need to know the optimal metric; in order to obtain the optimal metric, we need the active set. The practical solution to this dilemma is to find a good approximation of the optimal metric. Because both square tiling and the circle packing are the discrete analogies of the extremal length of a smooth quadrilateral, the metric (circle radii) induced by the circle packing method is a good initial guess of the optimal metric. In our algorithm pipeline, we compute the circle packing metric using the Ricci flow method as the first step.

4.1. Circle Packing Metric

Discrete Ricci flow generalizes surface Ricci flow to the discrete setting. Suppose $M=(V,E,F)$ is a polyhedral surface, each face is a Euclidean triangle, and V, E, F represents the set of vertices, edges and faces respectively. A *discrete Riemannian metric* of M is represented as the edge length function $l: E\to\mathbb{R}^+$, which satisfies the triangle inequality on each face. The *discrete Gaussian curvature* is defined as the angle deficit on each vertex,

$$K(v_i)=\begin{cases} 2\pi-\sum_{jk}\theta_i^{jk} & v_i\notin\partial M \\ \pi-\sum_{jk}\theta_i^{jk} & v_i\in\partial M \end{cases} \tag{8}$$

where θ_i^{jk} is the corner angle at v_i in triangle $[v_i,v_j,v_k]$. The total curvature satisfies the Gauss-Bonnet theorem

$$\sum_{v_i\in V}K(v_i)=2\pi\chi(M).$$

We associate each vertex v_i with a disk centered at v_i with radius r_i and define the edge length as $l([v_i,v_j])=r_i+r_j$, which is called the *tangential circle packing metric*. The *discrete Ricci flow* is given by

$$\frac{dr_i}{dt}=(\bar{K}(v_i)-K(v_i))\,r_i,$$

In fact, this is the gradient flow of a convex energy, the so-called entropy energy. Let $u_i=\log r_i$, $\mathbf{u}=(u_1,u_2,\cdots,u_n)$. The entropy energy is given by

$$E(\mathbf{u})=\int_{\mathbf{0}}^{\mathbf{u}}\sum_{i=1}^{n}(\bar{K}(v_i)-K(v_i))du_i.$$

The gradient of the entropy energy is

$$\nabla E(\mathbf{u})=(\bar{K}_1-K_1,\bar{K}_2-K_2,\cdots,\bar{K}_n-K_n)^T. \tag{9}$$

Suppose one interior edge $[v_i,v_j]$ is adjacent to two faces $[v_i,v_j,v_k]$ and $[v_j,v_i,v_l]$, the in-circle radii of them are τ_k and τ_l respectively, then the edge weight is

$$w_{ij}:=\frac{\tau_k+\tau_l}{r_i+r_j}.$$

If the edge is on the boundary, and only adjacent to one face $[v_i,v_j,v_k]$, then $w_{ij}:=\frac{\tau_k}{r_i+r_j}$.

The Hessain matrix of the energy is given by

$$\frac{\partial^2 E(\mathbf{u})}{\partial u_i\partial u_j}=\begin{cases} 0 & i\neq j, v_i\not\sim v_j \\ w_{ij} & i\neq j, v_i\sim v_j \\ -\sum_k w_{ik} & i=j \end{cases} \tag{10}$$

The energy is concave on the space $\{\mathbf{u}|\sum_i u_i=0\}$, and can be optimized using Newton's method efficiently.

First, we set the circle radius for each vertex equal to 1, namely $u_i=0$ for all $v_i\in G$. Then, we set the target curvature to be 0 for all interior vertices, $\frac{\pi}{2}$ for the four corners, and 0 for all other boundary vertices. We optimize the concave energy $E(\mathbf{u})=\int_{\mathbf{0}}^{\mathbf{u}}\sum_i(\bar{K}_i-K_i)du_i$ using Newton's method. Details can be find in Alg. 1.

4.2. Square Tiling Metric

The circle packing metric gives an approximation to the optimal square tiling metric. At each step, we use the current metric to compute all the shortest paths connecting the left side and the right side, and denote their union as Γ. Then we solve the problem of

Algorithm 1 Circle Packing Metric

Require: A planar graph G with a boundary B and four corner nodes $\{v_0, v_1, v_2, v_3\} \subset B$
Ensure: A circle packing metric of G

Set the circle radius to be 1, $u_i = 0, \forall v_i \in G$.
Set the target curvature for the four corner nodes to be $\pi/2$, and 0 for others.
repeat
  Compute the edge lengths $l_{ij} = e^{u_i} + e^{u_j}$.
  Compute the corner angles using cosine law

$$\cos\theta_i^{jk} = \frac{l_{ij}^2 + l_{ki}^2 - l_{jk}^2}{2l_{ij}l_{ki}}.$$

  Compute the Gaussian curvature at each vertex using Eqn. 8
  Compute the gradient ∇E of the entropy energy using Eqn. 9
  Compute the Hessian matrix H of the entropy energy using Eqn. 10
  Solve the linear system $H\delta\mathbf{u} = \nabla E(\mathbf{u})$.
  Update the conformal factor $\mathbf{u} \leftarrow \mathbf{u} + \varepsilon\delta\mathbf{u}$, where ε is the step-length parameter.
until the norm of the gradient $\nabla E(\mathbf{u})$ is less than a threshold .

quadratic programming with convex constraints by using the active set method to get an updated metric. We repeat this procedure until the updated metric and the previous metric are very close to each other. Details can be found in Alg. 2.

Algorithm 2 Square Tiling Metric

Require: A planar graph G with a boundary B and four corner nodes $\{v_0, v_1, v_2, v_3\} \subset B$
Ensure: A Square tiling of G on the plane

The corner nodes divide the boundary into 4 segments $\{s_0, s_1, s_2, s_3\}$.
Use tangential circle packing method to compute a circle packing, which induces the initial metric ρ on G.
$\Gamma \leftarrow \emptyset$
repeat
  for all $v_i \in s_0$ **do**
    Compute the shortest path from v_i to any vertex on s_2, denote the path as γ_i.
    $\Gamma \leftarrow \Gamma \cup \gamma_i$
  end for
  Solve the convex quadratic programming problem to obtain the updated metric ρ.
until The change of the metric ρ is less than a small threshold
Replace each node v_i by a square with edge length ρ_i, and pack the squares using graph connectivity.

4.3. Packing

After the optimal metric ρ has been obtained, we need pack the squares to fill a rectangle. The packing algorithm is based on the combinatorics of the graph, which utilizes the fact that the graph is triangulated. The first step is to lay out the squares on the boundary of the rectangle, which can be calculated without any ambiguity. The computational algorithm has been shown in Alg.3. The interior arrangement is more complicated than the boundary case. Given a triangle with three vertices sorted counter-clock-wisely, the corresponding squares are denoted as L, M, N respectively, where L and M have been laid out in advance. By analyzing the positions and connectivity, the packing scenario can be classified into four categories as shown in7. In the following discussion, we denote the coordinates of lower left corner of each square as (x, y) and the edge length of the square as r. More explicitly, the coordinates of lower left corner of L, M, N are represented as $(L_x, L_y), (M_x, M_y), (N_x, N_y)$ respectively, and the corresponding edge lengths are represented as L_r, M_r, N_r. All squares are packed up in the law of Alg.4. Fig.1 demonstrates the computational result for a given graph in colorful images.

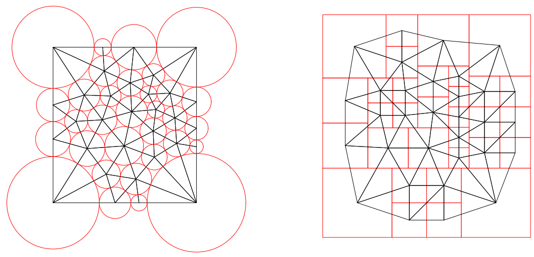


Figure 6: *Schematic description: planar graph embedding induced by two types of geometric objects: circle packing and square tiling. Circle packing can avoid degeneracy in circles, but causes the gap in the derived layout. Square tilling may generate trivial squares without additional treatment, but makes efficient use of display region and produces the regular layout.*

Algorithm 3 Boundary Square Tiling

Require: A planar graph G with a boundary B and four corner nodes $\{v_0, v_1, v_2, v_3\} \subset B$, and the metric $\rho : V \rightarrow R^+$.
Ensure: A Square tiling of the boundary vertices of B on the plane

Set the lower-left corner of v_0 to be $(0,0)$.
The path along S_0 is $\{v_{i_0}, v_{i_1}, v_{i_2}, \cdots, v_{i_k}\}$ sorted counter-clockwisely, where $v_{i_0} = v_0$ and $v_{i_k} = v_1$, $l_j = \sum_{t=0}^{j-1} \rho(v_{i_t})$. Set the lower-left corner of v_{i_j} as $(l_j, 0)$ and denote the lower-left corner of v_1 as $(a, 0)$.
The path along S_1 is $\{v_{i_0}, v_{i_1}, v_{i_2}, \cdots, v_{i_l}\}$ sorted counter-clockwisely, where $v_{i_0} = v_1$ and $v_{i_l} = v_2$, $l_j = \sum_{t=0}^{j-1} \rho(v_{i_t})$,set the upper-right corner of v_{i_j} as (a, l_j). Denote the upper-right corner of v_2 as (a, b).
The path along S_2 is $\{v_{i_0}, v_{i_1}, v_{i_2}, \cdots, v_{i_m}\}$ sorted counter-clockwisely, where $v_{i_0} = v_2$ and $v_{i_m} = v_3$, $l_j = a - \sum_{t=0}^{j-1} \rho(v_{i_t})$, set the upper-right corner of v_{i_j} as (l_j, b).
The path along S_3 is $\{v_{i_0}, v_{i_1}, v_{i_2}, \cdots, v_{i_n}\}$ sorted counter-clockwisely, where $v_{i_0} = v_3$ and $v_{i_n} = v_0$, $l_j = b - \sum_{t=0}^{j-1} \rho(v_{i_t})$, set the upper-left corner of v_{i_j} as $(0, l_j)$.

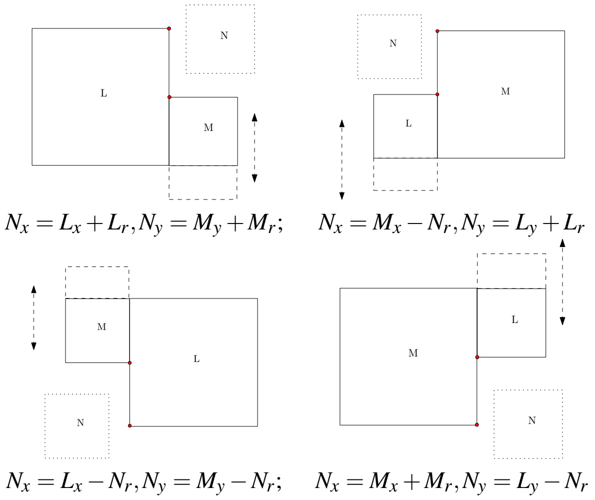


Figure 7: *Three counter-clockwise vertices in a triangle are represented as squares L, M, N respectively. There are four cases of position for square N, which dependents on L and M.*

Algorithm 4 Square Tiling

Require: A planar graph G with a boundary B and four corner nodes $\{v_0, v_1, v_2, v_3\} \subset B$, and the metric $\rho : V \to R^+$.
Ensure: A Square tiling of G on the plane

```
Pack the squares of all boundary vertices using algorithm 3.
repeat
    for all face f = [L, M, N] ∈ G do
        if L, M are packed and N is not packed then
            if N is packable then
                Pack N
            end if
        end if
    end for
until All vertices have been packed
```

---

5. Experimental Results

In this section, we make use of square tiling method for several practical applications. All experiments are conducted on a Windows 7 64 bit platform, with a single 2.90 GHz Intel CPU, 12 GB RAM memory. All the algorithms have been developed in generic C++, and compiled using Visual Studio 2013. The quadratic programming is solved using the QuadProg++ library [Gas13], which is a C++ implementation of the Goldfarb-Idnani dual active set algorithm for quadratic programming problems [GI83].

5.1. Applications of Square Tilling

Arranging geometric objects in a two-dimensional space with overlap removal is investigated in many different contexts, such as, optimization community for bin packing and graph drawing by using alternative contact graph to represent planar node-link graph. The overlap removal problem also appears in the context of information visualization as a mechanism to build the layout. We will apply our square tilling method to produce a novel overlap removal algorithm for the problem of arranging squares in the visual space. Fig.8 shows a colorful USA map with license plates attached to each state. Due to regional size discrepancy, there may be some visual clutter in a simply mixed image. Boxes containing plates overlap considerable hamper the visualization. We will apply the proposed method to design a novel layout for plates image. Firstly, we take the skeleton of map into account, where each state except for ALASKA and HAWAII is represented by a node and adjacency between states is depicted by node-link structure. To avoid degenerate squares in the skinny layout, some extra nodes are manually added into the skeleton, including sea points close to MICHIGAN, MAINE, NEW JERSEY, VIRGINIA, FLORIDA. With fixing four boundary points at MAINE, WASHINGTON, CALIFORNIA, and sea point near to FLORIDA, such triangle skeleton is topologically equivalent to a quadrilateral. The square tiling method is able to map the derived quadrilateral mesh into a planar rectangle tiled by varying-size squares. For better visual aesthetics, we scale up the width of rectangle to change nested squares into nested small rectangles so as to match actual rectangular icons of plates. Then icons of plates are artificially embedded into small rectangular boxes which have a combinatorial structure of still preserving neighborhood. We focus on generating a regular layout with particularly compact and overlap removal features, so the sizes of squares may not be proportional to the actual regional areas. Vacant regions in the plate image are corresponding to extra sea points meanwhile plates of ALASKA and HAWAII are arranged into the regular layout for complete representation. Notice that the layout is easy to read, mainly due to the effective use of display area. Moreover, it is not difficult to note that the overlap removal structure makes the visual identification of nearby objects an easier task.

The second application concerns multi level structures visualization. A graph composed of several levels can be represented in the form of large size squares tiled by smaller size squares. Large size squares represent the upper levels while nested small squares correspond to lower levels. Fig.9 illustrates a nested representation of the Trans-American Passenger Network. The whole network consists of 11 colored regions, Cascadia, Great lakes, Northeast, NorCal, Front Range, Socal, Arizona Sun Corridor, Texas Triangle, Gulf Coast, Piedmont, and Florida. Each region includes several lower level stations in the transportation line. We group sibling stations and utilize the proposed square tiling method to represent the whole network into a rectangle tiled by 11 different squares, then subdivide these squares according to the ongoing relations.

Our last application regards conformal surface parameterization, which is the process of mapping a surface to a planar domain. Conformal surface parameterizations have advantages on preserving angular structure, being intrinsic to geometry, stable with respect to different triangulations and small deformations. It is desirable to find the global conformal surface parameterization without any seams. The early work of global conformal parameterization has been done in [GY02, GY03], but the global conformal parameterization is non-unique. The following paper introduces an explicit method based on certain holomorphic differential forms to find the optimal global conformal parameterizations of arbitrary surfaces and designs the metrics for measuring the quality of conformal parameterizations [JWYG10]. More powerful tool for conformal ge-

Figure 8: *Geographic information visualization by square tiling method: Left frame is the colorful USA map with a attached triangular skeleton, where each state is represented as a node except ALASKA and HAWAII; Right frame is the license plates image in a structured overlap free layout, which is generated by mapping the abstract skeleton with fixed four boundary points, MAINE, WASHINGTON, CALIFORNIA, and sea point near to FLORIDA, into a rectangle tiled by varying size squares. For benefitting visual aesthetics, we scale up the width of rectangle to change nested squares into nested smaller rectangles for matching actual rectangular icons of plates. Later, state plates are mutually inserted into those boxes. Vacant regions correspond to artificial sea points which are introduced for avoiding degenerate squares in the rectangular layout.*

ometry is the Ricci flow-based method which designs any Riemannian metric by user-defined curvature [WDXD10]. For example, Fig.2 takes the Ricci flow method to convert a 3D facial surface into a 2D domain. There is a simple topology for the input facial surface, such as a simply connected domain. Choosing four points on the boundary v_0, v_1, v_2, v_3 and setting the target curvature to be $\frac{\Pi}{2}$ at v_i and zero everywhere else, the Euclidean Ricci flow will conformally map this surface onto a square. Our proposed conformal parameterization method has no need to set specific fixed curvatures at corner points in resulting 2D domain. The extremal length theory can inherently map a topological quadrilateral into an unique rectangle tiled by squares, whose edge lengths equal to node weights obtained through the solution of Eqn.3. Fig.10 demonstrates the square tiling representation for two 3D facial surfaces. It is easy to note that face shape information is preserved by the combinatorial structure of squares.

5.2. Efficiency Improvement

We use the circle packing metric to approximate the optimal square tiling metric, which can greatly improve the computational efficiency. Tables 1 and 2 show the difference of the running time on the same set of graphs. When the complexity of the graph increases, the possible paths from the left side to the right side will increase exponentially. Therefore, direct quadratic programming becomes prohibitively expensive. For large graphs, by applying circle packing, the computational time has been reduced significantly.

# Vertex	# Face	# Edge	# Path	# Iterations	Time
54	84	137	42	6	78ms
102	171	272	90	9	204ms
214	377	590	182	17	3.057s
307	556	862	272	32	65.26s
401	731	1131	357	97	1498.707s

Table 1: *Square tiling for a quadrilateral graph(without circle packing)*

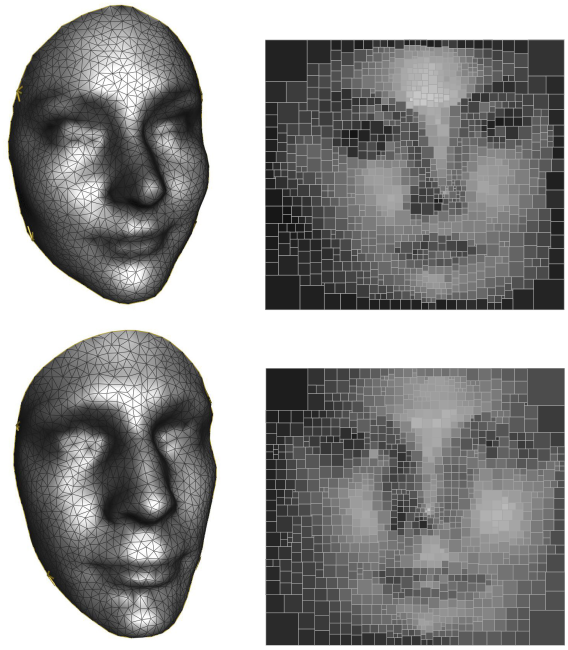


Figure 10: *Square tiling method for the planar representations of two 3D triangle meshes, which are topologically equivalent to planar quadrilaterals. The left column is the original 3D triangular representations of two faces; the right column is the planar combinatorial representations of two faces, which are constructed by varying size squares in contact with disjoint regions.*

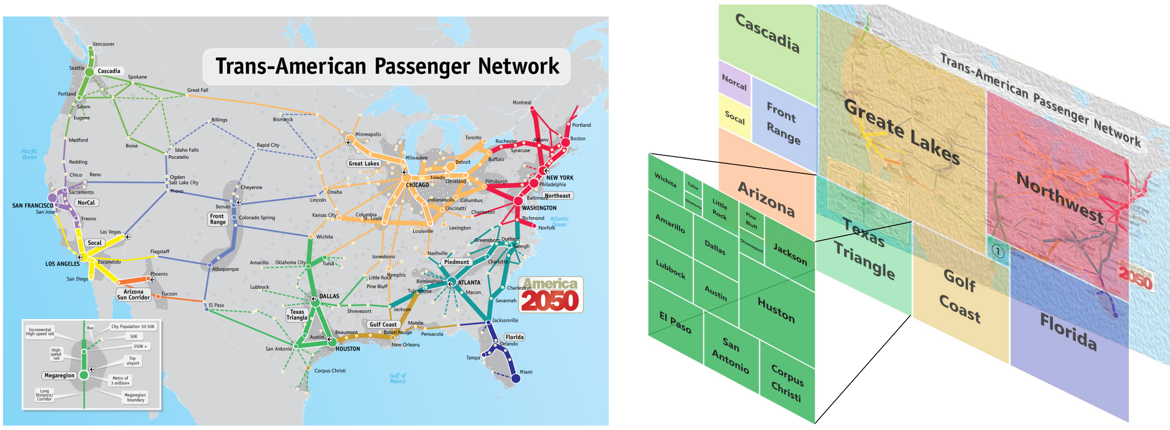


Figure 9: *The Trans-American Passenger Network is represented by a square tiling. Each regional network is mapped to a color square labeled with the name of the regional network. The detail of subnetwork is reprented by suqare tiling, recursivly.*

# Vertex	# Face	# Edge	# Path	# Iterations	Time
54	84	137	42	5	74ms
102	171	272	90	9	215ms
214	377	590	182	10	1.189s
307	556	862	272	24	37.784s
401	731	1131	357	53	443.866s

Table 2: *Square tiling for a quadrilateral graph(with circle packing)*

6. Conclusion

This work proposes a novel graph embedding method by square tiling, which provides a planar embedding of the topological quadrilateral through extremal length theory in conformal geometry. The process is to compute the nodes weights in use of quadratic programming with convex linear constraints and take these weight values as edge lengths of squares, then stitching these squares in the way of preserving relations between nodes. This resulting planar representation takes a square to represent a node and depicts the link relation as the tangency relation between squares. We take the novel method to build regular geometric layouts to visualize data, in where several visual requirements are realized simultaneously, including structured overlap-free arrangement, compact use of display area, neighborhood preservation. For improving the computational efficiency, the circle packing graph embedding method based on discrete Ricci flow is proposed to approximate the optimal metric. Due to the emergency of degenerate squares which shrink into points in the final layout, we have also introduced an adding points modification method, for example, the vacant regions in the state plate image are corresponding to the extra nodes as shown in Fig.8. We can also make use of square tiling method for multi level structure replacement. Fig.9 illustrates the nested structure for two level Trans-American Passenger Network. There is no doubt that square tiling method can be generated for more deep structures. Furthermore, the proposed method originates from conformal geometry, so it can be viewed as another tool for conformal surface parameterization. A practical method to compute the conformal structure of a topological quadrilateral are utilized to map 3D facial surfaces to 2D rectangular images. Fig.10 shows that the extremal length computed in this paper are invariant under conformal transformation group.

In the future, we plan on investigating possible methods to accelerate the process for more large data. We are also interested in extending this work to represent closed surfaces in square tiling method, such as torus.

References

[BGPV05] Buchsbaum A. L., Gansner E. R., Procopiuc C. M., Venkatasubramanian S.: Rectangular layouts and contact graphs. *Acm Transactions on Algorithms Talg Homepage 4*, 1 (2005), 607–616. 2

[BHVW00] Bruls M., Huizing K., van Wijk J. J.: *Squarified treemaps*. Springer, 2000. 3

[Bro40] Brooks R.: The dissection of rectangles into squares by r. l. brooks, c. ab smith, a. h. stone and w. t. tutte. 2

[Can94] Cannon J. W.: The combinatorial riemann mapping theorem. *Acta Mathematica 173*, 2 (1994), 155–234. 2, 6

[CFP94] Cannon J. W., Floyd W. J., Parry W. R.: Squaring rectangles: the finite riemann mapping theorem. *Contemporary Mathematics 169* (1994), 133–133. 2

[CLY09] Chuang J. H., Lin C. C., Yen H. C.: Drawing graphs with nonuniform nodes using potential fields. *Journal of Visual Languages and Computing 20*, 6 (2009), 385–402. 3

[DMS06] Dwyer T., Marriott K., Stuckey P. J.: Fast node overlap removal âĂŤ addendum. *Proc.int.symp.on Graph Drawing* (2006), 153–164. 3

[Fel12] Felsner S.: *Rectangle and Square Representations of Planar Graphs*. Springer New York, 2012. 2

[FP02] Fekete J. D., Plaisant C.: Interactive information visualization of a million items. *Craft of Information Visualization* (2002), 279–286. 3

[Gas13] Gaspero L. D.: Quadprog++ library(version 1.2.1). http://quadprog.sourceforge.net, Jan. 2013. 9

[GCM*16] GOMEZNIETO E., CASACA W., MOTTA D., HARTMANN I., TAUBIN G., NONATO L.: Dealing with multiple requirements in geometric arrangements. *IEEE Transactions on Visualization and Computer Graphics 22*, 3 (2016), 1223–1235. 3

[GH09] GANSNER E. R., HU Y.: Efficient node overlap removal using a proximity stress model. In *Graph Drawing* (2009), pp. 206–217. 3

[GI83] GOLDFARB D., IDNANI A.: A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical programming 27*, 1 (1983), 1–33. 9

[GNCMH16] GOMEZ-NIETO E., CASACA W., MOTTA D., HARTMANN I.: Dealing with multiple requirements in geometric arrangements. *IEEE Transactions on Visualization and Computer Graphics 22*, 3 (2016), 1223–1235. 1

[GY02] GU X., YAU S. T.: Ieee transactions on pattern analysis and machine intelligence. *Communications in Information and Systems 2*, 2 (2002), 121–146. 9

[GY03] GU X., YAU S. T.: Global conformal parameterization. In *First Eurographics Symposium on Geometry Processing, Aachen, Germany, June 23-25, 2003* (2003), pp. 127–137. 9

[HK98] HOUSE D. H., KOCMOUD C. J.: Continuous cartogram construction. In *Visualization '98. Proceedings* (1998), pp. 197 – 204. 2

[HLSG10] HUANG X., LAI W., SAJEEV A. S. M., GAO J.: A new algorithm for removing node overlapping in graph visualization. *Information Sciences 177*, 14 (2010), 2821–2844. 3

[JS91] JOHNSON B., SHNEIDERMAN B.: Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Visualization, 1991. Visualization '91, Proceedings., IEEE Conference on* (1991). 3

[JWYG10] JIN M., WANG Y., YAU S. T., GU X.: Optimal global conformal surface parameterization. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (2010), pp. 127–137. 9

[KK88] KOZMINSKI K. A., KINNEN E.: Rectangular dualization and rectangular dissections. *IEEE Transactions on Circuits and Systems 35*, 11 (1988), 1401–1416. 2

[KMN12] KOBOUROV S. G., MONDAL D., NISHAT R. I.: *Touching triangle representations for 3-connected planar graphs*. Springer Berlin Heidelberg, 2012. 2

[Koe36] KOEBE P.: Kontaktprobleme der konformen abbildung. 141–164. 2

[LL84] LEINWAND S. M., LAI Y. T.: An algorithm for building rectangular floor-plans. In *Design Automation, 1984. 21st Conference on* (1984), pp. 663 – 664. 2

[MK16] MARINO J., KAUFMAN A.: Planar visualization of treelike structures. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 1–1. 2

[MSTH03] MARRIOTT K., STUCKEY P., TAM V., HE W.: Removing node overlapping in graph layout using constrained optimization. *Constraints 8*, 2 (2003), 143–171. 3

[Rai34] RAISZ E.: The rectangular statistical cartogram. *Geography Review 24*, 2 (1934), 292–296. 2

[Sch93] SCHRAMM O.: Square tilings with prescribed combinatorics. *Israel Journal of Mathematics 84*, 1-2 (1993), 97–118. 2, 6

[Shn13] SHNEIDERMAN B.: Treemaps for space-constrained visualization of hierarchies [online]. Available: http://www.cs.umd.edu/hcil/treemap-history/, 1998âĂŞ2013. 3

[SSS*12] STROBELT H., SPICKER M., STOFFEL A., KEIM D., DEUSSEN O.: Rolled-out wordles: A heuristic method for overlap removal of 2d data representatives. *Computer Graphics Forum 31*, 3pt3 (2012), 1135–1144. 3

[SW01] SHNEIDERMAN B., WATTENBERG M.: Ordered treemap layouts. In *infovis* (2001), IEEE, p. 73. 3

[Tob04] TOBLER W.: Thirty five years of computer cartograms. *Annals of the Association of American Geographers 94*, 1 (2004), 58–73. 2

[WDXD10] WEI Z., DIMITRIS S., XIANFENG DAVID G.: Ricci flow for 3d shape analysis. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE 32*, 4 (2010), 662–677. 10

[YS93] YEAP K. H., SARRAFZADEH M.: Floor-planning by graph dualization: 2-concave rectilinear modules. *Siam Journal on Computing 22*, 3 (1993), 500–526. 2