

Exploring Circle Packing Algorithms*

Kevin Pratt¹, Connor Riley², and Donald R. Sheehy³

1 University of Connecticut, Storrs, USA

kevin.pratt@uconn.edu

2 University of Connecticut, Storrs, USA

connor.riley@uconn.edu

3 University of Connecticut, Storrs, USA

don.r.sheehy@gmail.com

Abstract

We present an interactive tool for visualizing and experimenting with different circle packing algorithms. The source code can be found at: <https://github.com/interl0per/CirclePacking>

1998 ACM Subject Classification F.2.2 [Nonnumerical Algorithms and Problems] Geometrical problems and computations

Keywords and phrases Computational Geometry, Processing, Javascript, Visualization, Incremental Algorithms

Digital Object Identifier 10.4230/LIPIcs.SoCG.2016.69

Category Multimedia Contribution

1 Introduction

The Koebe Embedding Theorem provides a wonderful link between questions of geometry, topology, combinatorics, and complex analysis. It states that every planar graph can be realized as the intersection graph of a collection of interior disjoint circles in the plane. That is, the vertices correspond to circles and two circles are tangent if and only if their corresponding vertices share an edge. Such a graph representation is called a circle packing. They have found application in computational geometry such as in a proof of the planar separator theorem [4] or for bounding eigenvectors of graph Laplacians [3].

Technically, circle packings exist for any planar graph, but it is often simpler algorithmically to work with 3-connected, triangulated planar graphs and we will assume such graph throughout. There are several known algorithms for computing a circle packing from a given planar graph. Early proofs of the existence of circle packings were non-constructive, but the work of Colin de Verdiere[2] and others [5, 1] showed that circle packings can be computed by convex optimization, ushering in several different iterative algorithms. A good introduction to the mathematical and algorithmic aspects of circle packings can be found in the book by Stephenson [6].

As shown in the work of Bobenko and Springborn, there are several different energy functionals that one could minimize to arrive at a circle packing [1]. However, there are simple heuristics that are not known to give correct algorithms, yet still seem to produce good circle packings. We present a tool designed to make it easy to mix and match circle

* This work was partially supported by the National Science Foundation under grants CCF-1525978 and CCF-1464379.



© Kevin Pratt, Connor Riley, Donald R. Sheehy;

licensed under Creative Commons License CC-BY

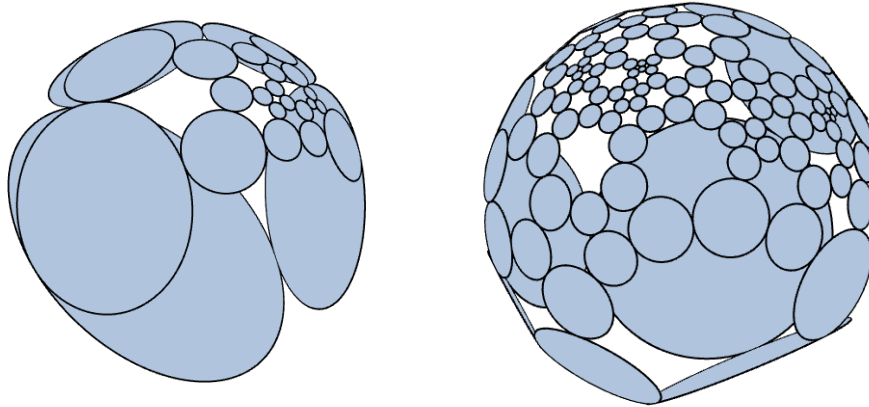
32nd International Symposium on Computational Geometry (SoCG 2016).

Editors: Fekete and Anna Lubiw; Article No. 69; pp. 69:1–69:4

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



■ **Figure 1** Stereographic projection of a circle packing.

packing heuristics. One can quickly generate a triangulated planar graph and then apply different heuristics to move it towards a circle packing representation.

2 Background

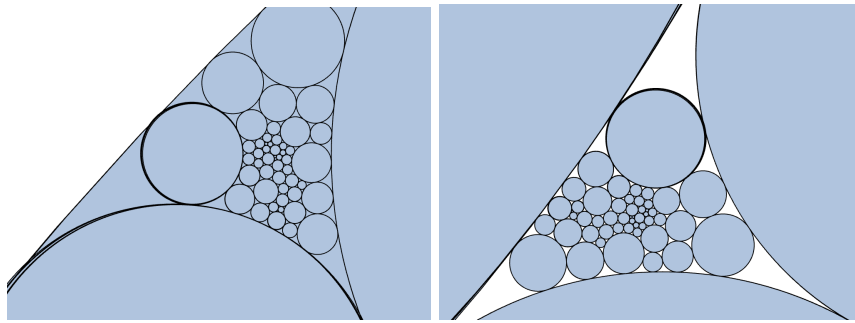
Weighted Delaunay Triangulation. Circle packings are closely related to several other well-known topics in computational geometry. The most notable, being *weighted Delaunay triangulations*. This is the dual to the so-called power diagram, a Voronoi diagram on the points where the (squared) distance to a circle c with center p and radius r is defined to be $\pi_c(x)^2 := \|x - p\|^2 - r^2$. A circle packing representation of a graph realizes the graph as the weighted Delaunay triangulation where the vertices are the centers of the disks and the weights are the radii.

Equilibrium Stresses. A *stress* is an assignment of real numbers to edges that may be interpreted as spring constants. By Hook's Law, the magnitude of the force exerted by a spring is the spring constant times its length. A stress is an *equilibrium stress* if for each vertex, the sum of the forces exerted on it by all its incident edges is exactly zero.

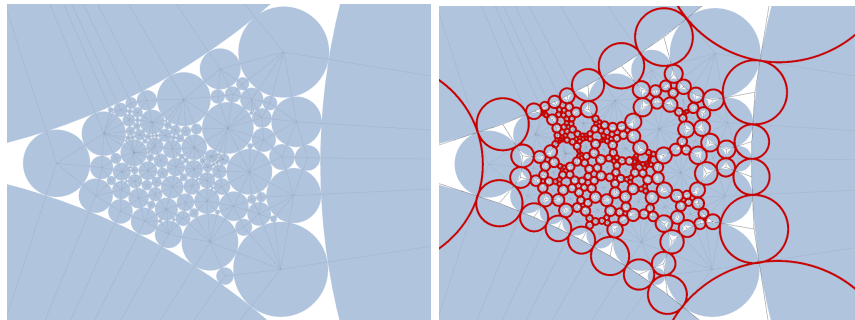
Weighted Delaunay triangulations have another interpretation as projections of convex polyhedra in \mathbb{R}^3 . The Maxwell-Cremona Correspondence implies that projections of convex polyhedra always admit an equilibrium stress that is nonnegative on all edges except those of the convex hull. Thus, an alternative (dual) perspective on circle packings is that rather than looking for the correct radii, one could look for the correct equilibrium stress. Given a stress, one finds the embedding by Tutte's algorithm. This involves solving one system of linear equations derived from the graph.

Stereographic Mappings and Möbius Transformations. Weighted Delaunay triangulations can also be realized as stereographic projections of convex polyhedra. In the case of circle packings this polyhedron is sometime called the Koebe polyhedron. It has the interesting property that the edges are all tangent to a sphere.

A Möbius Transformation of a plane can be obtained by performing the stereographic projection of the plane onto a sphere, then rotating or moving the sphere and then performing



■ **Figure 2** Two Möbius transformations of a circle packing.



■ **Figure 3** A packing and its dual.

the stereographic projection back onto the plane. Formally, a Möbius Transformation is a rational function defined on the extended complex plane $\hat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$ of the form:

$$f(z) = \frac{az + b}{cz + d} \quad (1)$$

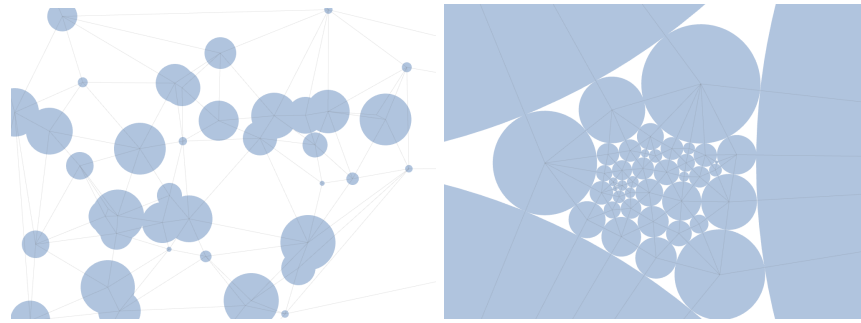
where $z \in \hat{\mathbb{C}}$ is a complex variable and $a, b, c, d \in \hat{\mathbb{C}}$ are complex numbers such that $ad - bc \neq 0$. Circle packings in the plane are unique up to Möbius Transformation.

Dual Packings. Circle packings of triangulations always admit a *dual packing* (Fig 3). By pressing the 'd' key, the user can see the dual packing.

3 Interactive Input

One immediate challenge to working interactively with circle packings is the need for input triangulations. We would like to be able to produce triangulations with a minimum of effort. The most natural approach is to draw circles and produce the weighted Delaunay triangulation of the circles. Input is given by clicking a center and dragging to set the radius of an input disk. As new points are added, we incrementally compute the weighted Delaunay triangulation. An advantage of this approach is that it is very easy to draw approximate circle packings. This allows one to experiment with examples where one has a clear idea of the expected output.

As a Möbius transformations will give new circle packings of the input graph, we allow the user to transform the plane to explore different packings. These transformations can be viewed by hitting the 'spacebar' key twice to change views and then moving the mouse.



■ **Figure 4** An input of circles (left) and the circle packing output (right).

If the user presses the 'spacebar' key again, then dragging with the mouse will instead rotate a 3D view of the circle packing mapped stereographically onto a sphere.

4 Algorithms

Currently we have two different algorithms implemented. The first is based on the algorithm from Stephenson [6]. Given any set $\{r_1, \dots, r_n\}$ of radii, it is possible to compute the angles of the triangles using the law of cosines. The final radii are those for which the angles at any vertex sum to exactly 2π . Thus, the algorithm searches for the radii of the disks by making small incremental updates to the radii, increasing the radius if the angle sum is more than 2π and decreasing the radius if the angle sum is less than 2π .

The second algorithm is based on spring embeddings of graphs such as Tutte's algorithm. In this algorithm, the embedded graph is interpreted as a system of springs. This "force-directed" algorithm is not known to terminate, but it seems to be quite effective on small examples.

5 Future Work

We are actively developing new features to provide both new ways to visualize the packings as well as new heuristics. The current implementation supports two of the major paradigms in circle packing algorithms. Next, we will consider algorithms based on discrete Ricci flow.

References

- 1 Alexander I. Bobenko and Boris A. Springborn. Variational principles for circle patterns and koebe's theorem. *Transactions of the American Mathematical Society*, 356(2):659–689, 2003.
- 2 Yves Colin de Verdière. Un principe variationnel pour les empilements de cercles. *Inventiones mathematicae*, 104(1):655–669, 1991.
- 3 Jonathan A. Kelner. Spectral partitioning, eigenvalue bounds, and circle packings for graphs of bounded genus. *SIAM J. Comput.*, 35(4):882–902, 2006.
- 4 Gary L. Miller, Shang-Hua Teng, William Thurston, and Stephen A. Vavasis. Separators for sphere-packings and nearest neighbor graphs. *J. ACM*, 44(1):1–29, 1997.
- 5 Bojan Mohar. A polynomial time circle packing algorithm. *Discrete Mathematics*, 117(1–3):257–263, 1993.
- 6 Kenneth Stephenson. *Introduction to Circle Packing*. Cambridge University Press, New York, New York, 2005.