International Journal of Computer & Information Technologies (IJOCIT)
www.ijocit.org & www.ijocit.ir
ISSN= 2345-3877

Hyperbolic Voronoi Diagram: A Fast Method

Zahra Nilforoushan *, Ali Mohadesm, Amin Gheibi, Sina Khakabi
Department of Computer Engineering[1,3,4] *, Mathematics and Computer Science*[2]
Kharazmi University[1]*, Amirkabir University of Technology*[2]*, Carleton University* [3]*, Simon Fraser University* [4]
Iran [1,2] *, Canada* [3.4]
shadi.nilforoushan@gmail.com [1], *mohades@aut.ac.ir* [2], *amin-gheibi@carleton.ca* [3], *sinakhm.cs84@aut.ac.ir* [4]

Abstract:- Voronoi diagrams have useful applications in various fields and are one of the most fundamental concepts in computational geometry. Although Voronoi diagrams in the plane have been studied extensively, using different notions of sites and metrics, little is known for other geometric spaces. In this paper, we present a simple method to construct the Voronoi diagram of a set of points in the Poincare hyperbolic disk, which is a 2-dimensional manifold with negative curvature. Our trick is to define and use some well-formed geometric maps which take care of connection between the Euclidean plane and Poincare hyperbolic disk. Finally, we give a brief report of our implementation.

Keywords: Computational geometry, Hyperbolic space, Geodesic, Voronoi diagrams.

1. Introduction

Voronoi diagrams for point-sets in d-dimensional Euclidean space E^d have been studied by a number of people in their original as well as in generalized settings. For a finite set $\boldsymbol{M} \subsetneq \boldsymbol{E^d}$, the (closest-point) Voronoi diagram of $\boldsymbol{M}$ associates each $\boldsymbol{p \in M}$ with the convex region $\boldsymbol{R(p)}$ of all points closer to p than to any other point in M. More formally, $\boldsymbol{R(p) = \{x \in E^d \mid d(x,}$

p) < d(x, q), $\forall$***q ϵ M – p },*** where d denotes the Euclidea distance function. Voronoi diagrams are of importance in a variety of areas other than computer science whose enumeration exceeds the scope of this paper (see for instance Aurenhammer's survey [3] or the book by Okabe, Boots, Sugihara and Chiu [18]). Shamos and Hoey [21] were the first to introduce the planar diagram to computational geometry and also demonstrated how to construct it efficiently. Using a dual correspondence to convex hulls discovered by Brown [5], its higher-dimensional analogues can be obtained using methods in Seidel [20].

As the variety of applications of the Voronoi diagram were recognized, people soon became aware of the fact that many practical situations are better described by some modification than by the original diagram. For example, diagrams under more general metrics [15, 16], for more general objects than points [9, 13], and of higher order [10, 14, 21] have been investigated.

The interesting properties of Voronoi diagrams attracted our attention to ask a natural question whether they will be satisfied in other spaces, especially for hyperbolic surfaces. Hyperbolic surfaces are characterized by negative curvature and cosmologists have suffered from a persistent misconception that negatively curved universe must be the finite 3-D hyperbolic space [23]. Although we do not see hyperbolic surfaces around us, often nevertheless nature does possess a few. For example, lettuce leaves and marine at worms exhibit hyperbolic geometry. There is an interesting idea about hyperbolic plane by ***W. P***. Thurston that if we move away from a point in hyperbolic plane, the space around that point expands exponentially [22]. Hyperbolic geometry has found applications in fields of mathematics, physics, and engineering. For example in physics, until we figure out whether or not the expansion of the universe is decelerating, hyperbolic geometry could be the most accurate way to define the geometries of fields. Einstein's invented his special theory of relativity based on hyperbolic geometry.

Now we switch to some applications of the Voronoi diagram in hyperbolic spaces. In [19] the authors deal with Voronoi diagram in simply connected complete manifolds with non- positive curvature, called Hadamard manifold. They proved that the facet of Voronoi diagram can be characterized by hyperbolic Voronoi diagram. They considered that these Voronoi diagrams and its dual structure, Delaunay triangulation, can be used as mesh generation, computer graphics and color space [6]. Another application of Voronoi diagram in hyperbolic models is triangulating a saddle surface, which is a part of the triangulation of a general surface. On general surface, some parts have positive curvature, other parts have negative curvature and other parts near zero. In such cases, one can divide the surface into some parts, make triangulation of each part according to their curvature.

Further applications of the Voronoi diagram in hyperbolic spaces are devoted to the Farey tessellation which is studied in [1]. The Teichmuller space for T^2 is the hyperbolic plane $\boldsymbol{H^2 = \{z = x+iy \in C \mid y > 0\}}$: $\boldsymbol{T^2_z}$ can be thought of as the quotient space of R^2 over the lattice $\boldsymbol{\{m.1+n.z \mid m, n \in Z\}} \subset \boldsymbol{C}$. Let $\boldsymbol{X} \subset \boldsymbol{H^2}$ be the set of all parameters z corresponding to the tori with three equally short shortest geodesics (i.e., tori glued from a regular hexagon). Then the Farey tessellation is nothing but the Voronoi diagram of $\boldsymbol{H^2}$ with respect to $\boldsymbol{X}$.

Such applications motivated us to study the Voronoi diagrams on hyperbolic spaces. In [17], the first two authors of this paper have studied the Voronoi diagram in Poincare hyperbolic disk where the running time of the proposed algorithm was $\boldsymbol{O(n^2)}$. In this paper, we present a new method to compute the Voronoi diagram in Poincare hyperbolic disk whose expected worst case running time is ***O(nlogn).***

This paper is organized as follows. In Section 2, a brief introduction to Poincare hyperbolic disk is studied. Section 3 briefly reports the required maps we used to transfer the Poincare hyperbolic disk to the Euclidean plane R^2, compute the Voronoi diagram in R^2 and then transfer it back. Section 4 is devoted to some Implementations.

2. Poincare hyperbolic disk

The Poincare hyperbolic disk is a two-dimensional model for hyperbolic geometry. Therefore it has a negative curvature and defined as the disk $D^2 = \{(x, y) \epsilon R^2 \mid x^2 + y^2 < 1\}$, with hyperbolic metric $ds^2 = \frac{dx^2+dy^2}{(1-x^2-y^2)^2}$. See [2] and [12] for details.

The Poincare disk is a model for hyperbolic geometry in which a geodesic (which is like a line in Euclidean geometry) is represented as an arc of a circle whose ends are perpendicular to the disks boundary (and diameters are also permitted). Two arcs which do not meet correspond to parallel rays, arcs which meet orthogonally correspond to perpendicular lines, and arcs which meet on the boundary are a pair of limit rays (see Figure 1).

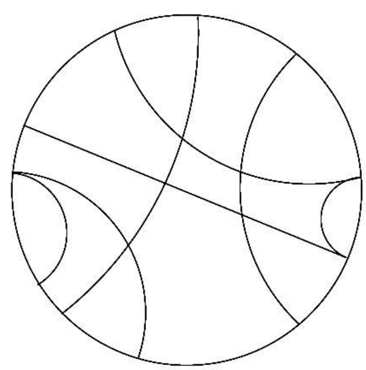

Figure 1: Poincaré disk and some of its geodesics

The equation of a geodesic of ***D2*** is expressed as either:

$$x^2 + y^2 - 2ax - 2by + 1 = 0, \quad \text{with} \quad a^2 + b^2 > 1,$$

or

$$ax = by.$$

Geodesics are basic building blocks for computational geometry on the Poincare disk. The distance of two points is naturally induced from the metric of D^2, consider two point $z_1(x_1, y_1)$, $z_2(x_2, y_2) \epsilon D^2$, the distance between z_1 and z_2, denoted by $d(z_1, z_2)$, can be expressed as

$$d(z_1, z_2) = \int_{the\ geodesic\ connecting\ z_1\ and\ z_2} ds = \tanh^{-1}\left(\left|\frac{z_2 - z_1}{1 - \overline{z_1} z_2}\right|\right).$$

3. Our method

Suppose we are given a set S of n points (representing sites) in D^2. To construct the Voronoi diagram, we use a combination of four maps to transfer these sites into the Euclidean plane. The maps are defined between four hyperbolic models and Euclidean plane, denoted by D^2, S^2, K^2, H^2 and R^2, respectively. In [7], Cannon et al. have an elegant discussion about these hyperbolic models:

1. $D^2 = \{(x, y) : x^2 + y^2 < 1\}$,
   $ds^2_{D^2} = \frac{dx^2 + dy^2}{(1 - x^2 - y^2)^2}$

2. $S^2 = \{(x, y, z) : x^2 + y^2 + z^2 = 1, z > 0\}$,
   $ds^2_{S^2} = \frac{dx^2 + dy^2 + dz^2}{z^2}$

3. $K^2 = \{(x, y) : x^2 + y^2 < 1\}$,
   $ds^2_{K^2} = 4\frac{dx^2 + dy^2}{(1 - x^2 - y^2)^2}$

4. $H^2 = \{(x, y, z) : z^2 - x^2 - y^2 = 1, z > 0\}$,
$ds^2_{H^2} = dx^2 + dy^2 - dz^2$.

The list of maps that we defined and used is given in the following:

(a) A central projection map from the point ***(0, 0, -1)***, $f_1 : D^2 \rightarrow S^2$ that

$$(x, y) \longmapsto (\frac{2x}{1+x^2+y^2}, \frac{2y}{1+x^2+y^2}, \frac{1-x^2-y^2}{1+x^2+y^2}).$$

(b) A lifting map $f_2 : S^2 \rightarrow K^2$ that

$$(x, y, z) \rightarrow (x, y, 1)$$

(c) A central projection map from the point ***(0, 0, 0)***, $f_3 : K^2 \rightarrow H^2$ that

$$(x, y, 1) \longrightarrow (\frac{x}{\sqrt{1-x^2-y^2}}, \frac{y}{\sqrt{1-x^2-y^2}}, \frac{1}{\sqrt{1-x^2-y^2}}).$$

(d) A central projection map from the point ***(0, 0, 2)***, $f_4 : H^2 \rightarrow R^2$ that

$$(x, y, z) \longrightarrow (\frac{-2x}{z-2}, \frac{-2y}{z-2})$$

Figure 2 is an illustration of the above mentioned spaces and the connecting maps.

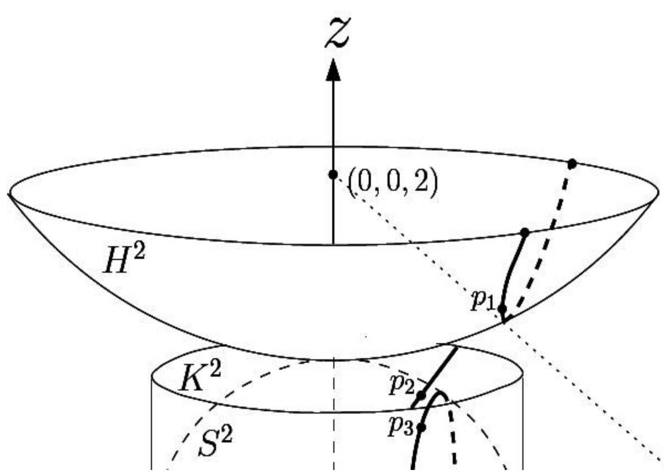

Figure 2: Spaces and the Connecting Maps

Now by using any algorithm in [4] for constructing the Voronoi diagram of the transferred sites in $\boldsymbol{R^2}$ which has the worst case running time complexity $\boldsymbol{O(nlogn)}$, the combination of the inverses of $\boldsymbol{f_i}$*'s* will allow us to obtain the Voronoi diagram in $\boldsymbol{D^2}$. This combination is robust, as the subsequent theorem verifies.

Theorem 1:

Let $\boldsymbol{z_1}$ and $\boldsymbol{z_2}$ be two points in $\boldsymbol{R^2}$ and $\boldsymbol{J}$ be their bisector. Then $\boldsymbol{f(J)}$ would be the bisector of $\boldsymbol{f(z_1)}$ and $\boldsymbol{f(z_2)}$ in $\boldsymbol{D^2}$ and $\boldsymbol{f = f_1^{-1} \ o\ f_2^{-1}\ o\ f_3^{-1}\ o\ f_4^{-1}}$ where $\boldsymbol{f_i}$ *'s* $\boldsymbol{(i = 1, 2, 3, 4)}$ are the above mentioned maps.

Proof:

Since we use the geodesics in each hyperbolic models and Euclidean plane $\boldsymbol{R^2}$, by using the corresponding metrics $\boldsymbol{d_s^2}$, we obtain that the bisector of two given points $\boldsymbol{z_1}$ and $\boldsymbol{z_2}$ in $\boldsymbol{R^2}$ will be mapped to the bisector of $\boldsymbol{f(z_1)}$ and $\boldsymbol{f(z_2)}$ in $\boldsymbol{D^2}$ and vice-versa.

As the complexity of the mentioned maps are linear, we conclude that the complexity of our method to compute the Voronoi diagram of a set of sites in $\boldsymbol{D^2}$ is $\boldsymbol{O(nlogn)}$ using any algorithm with the complexity $\boldsymbol{O(nlogn)}$ to compute the Voronoi diagram in $\boldsymbol{R^2}$ for the transferred sites from $\boldsymbol{D^2}$ and this yields the following main theorem of our paper.

Theorem 2:

Hyperbolic Voronoi diagram can be constructed with an $\boldsymbol{O(nlogn)}$ time complexity algorithm.

4. Implementation

In this section we present our implementation, and discuss its performance in some series of experiments, designed to test different aspects of our algorithm and implementation. Our code has been written in ***C++,*** and for visualization we have used ***MATLAB***. Our implementation with ***C++*** have three main steps: in the first step we transfer our points (sites) from Poincare disk to $\boldsymbol{R^2}$. In this step the program reads the coordinates of points from a file and then uses some methods and functions to transfer them to $\boldsymbol{R^2}$. In the second step we work on transferred points and use Fortune's algorithm and draw the Voronoi diagram of points. Source code of the Fortune's algorithm is available in [11, 24]. The output is the end points of the voronoi edges in $\boldsymbol{R^2}$. In the third step we transfer the end points to the Poincare disk. So we use the inverse mode of the maps defined in the first step. The output is the end points of Voronoi edges in Poincare disk. Since we have the formula for a geodesic in the Poincare disk, so we can draw Voronoi edges easily. We have used Visual ***C++*** in Microsoft Visual ***Studio.NET 2005*** with .***NET Framework 2.0*** and ***MATLAB Ra 2006***. All experiments run on an ASUS Notebook Z53 j series with 2.0 GHz core 2 duo CPU and 2 GB DDR2 RAM. In Figure 3 the result of our implemented method for five random sites situation is given.

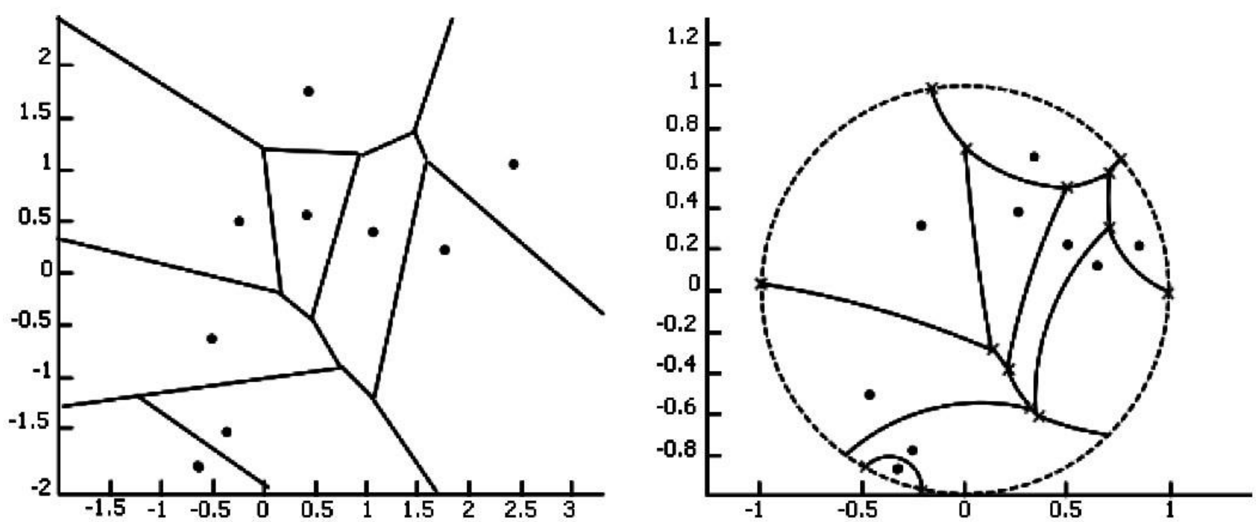

Figure 3: Result for the nine random sites in $\mathbb{R}^2$ and D^2

5. Acknowledgments

We would like to thank Professor Dr. R. Klein for reading the first version of the manuscript.

References

[1] S. Anisov, Geometrical spines of lens manifolds, Department of Mathematics, Utrecht University, 2005.
[2] J. W Anderson, Hyperbolic Geometry, New York. Springer-Verlag, 1999.
[3] F. Aurenhammer, Voronoi Diagrams: a Survey of a Fundamental Geometric Data Structure, ACM Computing Surveys
23(3) (1991), 345-405.
[4] F. Aurenhammer and R. Klein, Voronoi Diagrams, Handbook of Computational Geometry, J. Sack and G. Urrutia, editors, Elsevier Science Publishers, B.V. North-Holland, Chapter 5, pages: 201-290, 2000.
[5] K. Q. Brown, Voronoi diagrams from convex hulls, Inform. Process Lett. 9 (1979), 223-228.
[6] H. Brettel, F. Vienat, and J. D. Mollonl, Computerized simulation of color appearance for dichromats, Journal of Optical Society of America 14(10) (1997), 2647-2655.
[7] J. W. Cannon, W. J. Floyd, R. Kenyon, and W. R. Parry, Flavors of Geometry, MSRI Publications 31 (1997), 59-115.
[8] The CGAL User and Reference Manual: All Parts. Release 3.3., 2007.
[9] R. L. Drysdale and D. T. Lee, Generalization of Voronoi diagrams in the plane, SIAM J. COMPUT. 10 (1981), 73-87.
[10] H. Edelsbrunner, J. O'Rourke, and R. Seidel, Constructing arrangements of lines and hyperplanes with applications, Proc. 20th. Ann. IEEE Symp. FOCS (1983), 83-91.
[11] S. Fortune, http://cm.bell-labs.com/who/sjf/index.html.
[12] C. Goodman-Strauss, Compass and Straightedge in the Poincar e Disk, Disk. Amer. Math. Monthly 108 (2001), 33-49.
[13] D. G. Kirkpatrick, Efficient computation of continous skeletons, Proc. 20th. Ann. IEEE Symp. FOCS (1979), 18-27.
[14] D. T. Lee, On k-nearest neighbor Voronoi diagrams in the plane, IEEE Trans. Comp. C-31, 6 (1982), 478-487.
[15] , Two-dimensional Voronoi diagrams in the Lp metric, JASM 27(4) (1980), 604-618.
[16] D. T. Lee and C. K. Wong, Voronoi diagrams in L1(L1) metrics with two dimensional storage applications, SIAM J. COMPUT. 9 (1980), 200-211.
[17] Z. Nilforoushan and A. Mohades, Hyperbolic Voronoi Diagram, ICCSA 2006, LNCS 3984 (2006), 735 -742.
[18] A. Okabe, B. Boots, K. Sugihara, and N. Chiu, Spatial tesselations: concepts and applications of Voronoi diagrams, Wiley Series in Probability and Statistics, 2000.
[19] K. Onishi and J. Itoh, Voronoi diagram in simply connected complete manifold, IEICE TRANS. Fundamentals. E85-A, 5 (2002), 944-948.
[20] R. Seidel, A convex hull algorithm optimal for point sets in even dimensions, M. S. thesis, Rep. 81-14, Dep. Computer Science, Univ. of British Colombia, 1981.
[21] M. I. Shamos and D. Hoey, Closest-Point Problems, Proceedings 16th IEEE Symposium on Foundations of Computer Science (1975), 151-162.
[22] W. P. Thurston, Three dimensional Geometry and Topology, Princeton University Press, 1997.
[23] J. R. Weeks, The Shape of Space, CRC, 2nd edition, 2001.
[24] Voronoi Resources, http://www.skynet.ie/sos/mapviewer/voronoi.php.

International Journal of Computer & Information Technologies (IJOCIT)
Corresponding Author: *Zahra Nilforoushan*
August 23, 2013 Volume 1, Issue 1

Authors Profile:

Dr. Zahra Nilforoushan, is an Assistant Professor in the Department of Computer Engineering at Faculty of Engineering of Kharazmi University of Tehran. She completed her Ph.D. in Computational Geometry at Amirkabir University of Technology (Tehran Polytechnic) in 2009. Previously, she received her M.Sc. in Algebraic Geometry at Amirkabir University of Technology in 2002. Her main interests of research are Computational Geometry, Computer Graphics, Computer Vision, Differential Geometry, Projective Geometry, Hyperbolic Geometry and Coding.

****

Dr. Ali Mohades is an Associated Professor and manager of computer science group in faculty of mathematics and computer science in Amirkabir University of Technology in Tehran, Iran. He is the dean of laboratory of algorithms and computational geometry in faculty of mathematics and computer science. His main interest fields are computational geometry, robotics and facility location.

****

Amin Gheibi is currently a Ph.D. student at Carleton University, School of Computer Science. He received his M.Sc. in Computer Science at Amirkabir University of Technology in 2009. His main interest of research is Computational Geometry. However, he is interested in a wide range of topics in Computer Science and Mathematics including Statistical Pattern Recognition, Computer Vision and Image Processing.

****

Sina Khakabi is currently a Software Development Engineer (SDE) at Microsoft. He received his M.Sc. in Computer Science at Simon Fraser University in 2011. His main skills and expertise are Matlab, Python, Statistics, C++, Java, Machine Learning, HTML, Algorithms, Databases, PHP, JavaScript, Microsoft Office, Software Engineering, Programming, Linux, Latex and CSS.