Geodesics, Shortest Paths, and Non-Isomorphic Nets

Potvin Nicolas   Zolotov Boris

ULB,   November 2020

1   Introduction

Given a collection of 2D polygons, a *gluing* or a *net* describes a closed surface by specifying how to glue (a part of) each edge of these polygons onto (a part of) another edge. Alexandrov's uniqueness theorem [1] states that any valid gluing that is homeomorphic to a sphere and that does not yield a total facial angle greater than 2π at any point, corresponds to the surface of a unique convex 3D polyhedron (doubly covered convex polygons are also regarded as polyhedra). Note that the original polygonal pieces might need to be folded to obtain this 3D surface.

Unfortunately, the proof of Alexandrov's theorem is highly non-constructive. The only known approximation algorithm to find the vertices of this polyhedron [9] has a (pseudopolynomial) running time really large in n, where n is the total complexity of the gluing.

There is no known exact algorithm for reconstructing the 3D polyhedron, and in fact the coordinates of the vertices of the polyhedron might not even be expressible as a closed formula [8].

Enumerating all possible valid gluings is also not an easy task, as the number of gluings can be exponential even for a single polygon [5]. However one valid gluing can be found in polynomial time using dynamic programming [7, 10]. Complete enumerations of gluings and the resulting polyhedra are only known for very specific cases such as the Latin cross [6] and a single regular convex polygon [7].

The special case when the polygons to be glued together are all identical regular k-gons, and the gluing is *edge-to-edge* was studied recently for $k \geq 6$ [3] and $k = 5$ [2]. The aim of this project is to study the case of $k = 4$.

2   Nicolas' part: implementation of Chen—Han Algorithm

Chen—Han algorithm is presented in [4]. It is needed to build knowledge about the gluing. It has to be generalized from the case of the net of a polyhedron to an arbitrary gluing where polygons need not to be faces.

The API built to interface Chen & Han will be the core of this part and serve to demonstrate the algorithm through interactive examples coded in JavaScript. The aim is to propose an educational resource to explain this algorithm in an intuitive and visual manner.

3 Boris' part: enumeration of gluings, isomorphism check

Estimate of the number of vertices, enumeration of gluings was done in [2, 3].

4 Application Programming Interface

Both parts of the work will be connected through an API. Two main functions have been identified so far:

- Given a net, return the number of vertices that are part of the polyhedrons.

- Given a net, return a square matrix of dimensions $|V| \times |V|$. This matrix holds the distances between any two pairs of vertices, computed using the Chen & Han algorithm.

The format used to represent a net is an array of n arrays of k oriented line segments. The number n is arbitrary, but the number k is fixed to either 3 or 4 to handle respectively the classical situation where triangles are used and the the more specific situation of a net made of squares.

When gluing squares of the same size edge-to-edge, we can forget about the segments and only store the numbers that represent the equivalence class that the edge belongs to. The orientation in which to glue the edges and the net itself can be reconstructed from this data immediately, see Figure 1.
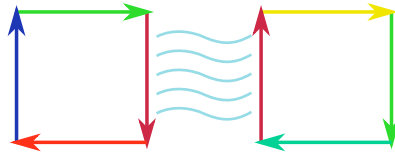

Figure 1: The information about two edges of squares belonging to the same equivalence class is enough to reconstruct the way they are glued.

4.1 Triangles and squares

When working on triangles, the Chen & Han algorithm is unchanged. The issue arising from the use of squares instead is that a geodesic can go through a single square more than once. Since the algorithm does not come back on a previously visited shape, this means that the algorithm needs to be slightly modified.

The idea is to run it longer to allow visiting again previously encountered squares. As proven in [11], the number of times a path goes through a single square is at most 7, which provides an upper bound on the number of steps of the modified algorithm.

References

[1] Alexandr Alexandrov. *Convex Polyhedra*. Springer-Verlag, Berlin, 2005.

[2] E. Arseneva, S. Langerman, and B. Zolotov. A complete list of all convex shapes made by gluing regular pentagons. In *XVIII Spanish Meeting on Computational Geometry*, page 1–4, Girona, Spain, 2019.

[3] Elena Arseneva and Stefan Langerman. Which Convex Polyhedra Can Be Made by Gluing Regular Hexagons? *Graphs and Combinatorics*, page 1–7, 2019.

[4] Jindong Chen and Yijie Han. Shortest paths on a polyhedron. In *6-th annual symposium on Computational geometry*, page 360–369, Berkley, California, USA, June 1990. SCG '90.

[5] Erik Demaine, Martin Demaine, Anna Lubiw, and Joseph O'Rourke. Enumerating foldings and unfoldings between polygons and polytopes. *Graphs and Combinatorics*, 18(1):93–104, 2002.

[6] Erik Demaine, Martin Demaine, Anna Lubiw, Joseph O'Rourke, and Irena Pashchenko. Metamorphosis of the cube. In *Proc. SOCG*, pages 409–410. ACM, 1999.

[7] Erik Demaine and Joseph O'Rourke. *Geometric folding algorithms*. Cambridge University Press, 2007.

[8] David Eppstein, Michael J Bannister, William E Devanny, and Michael T Goodrich. The Galois complexity of graph drawing: Why numerical solutions are ubiquitous for force-directed, spectral, and circle packing drawings. In *International Symposium on Graph Drawing*, pages 149–161. Springer, 2014.

[9] Daniel M Kane, Gregory N Price, and Erik D Demaine. A Pseudopolynomial Algorithm for Alexandrov's Theorem. In *WADS*, pages 435–446. Springer, 2009.

[10] Anna Lubiw and Joseph O'Rourke. When can a polygon fold to a polytope?, 1996.

[11] Boris Zolotov. Algorithmic Aspects of Alexandrov's Uniqueness Theorem. *Bachelor's Thesis, St. Petersburg State University*, 2019.