Triangulation Software: Computation in Cluster Algebra

Dorothée Grondin
Laval University

TRIANGULATION SOFTWARE

What is it about? - Cluster Algebras of Surfaces

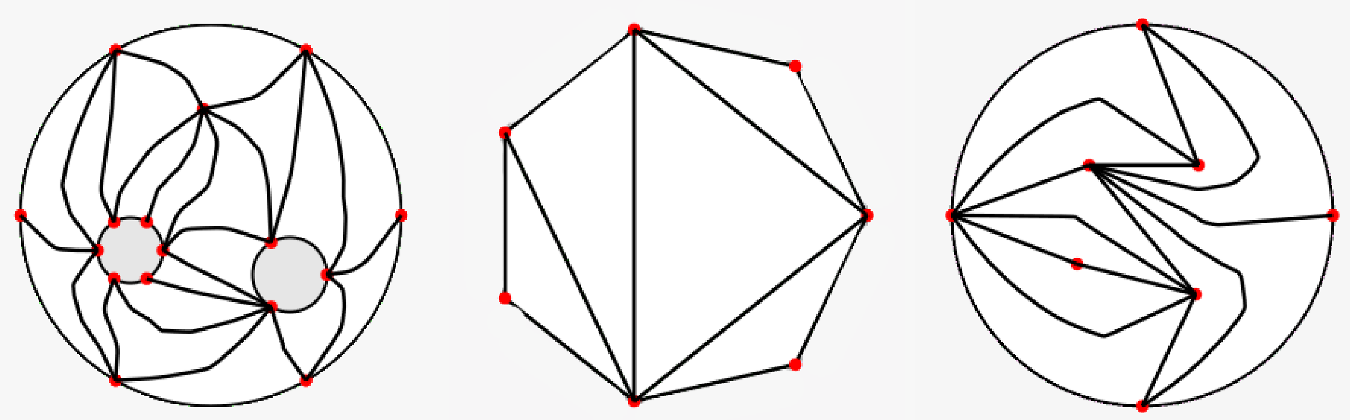
A cluster algebra $\mathcal{A}$(x,y,Q) is an algebra with a combinatorial structure. One starts with a set of generators $X = (x_1, x_2, ..., x_n)$ with x_i in the algebra. This set is called a cluster and its elements cluster variables. The combinatorial structure of the cluster algebra is given by an operation μ called mutation which takes one cluster variables x_i in the cluster X and exchange it with a new cluster variable x_i' giving us a new cluster X'.

There is a subset of cluster algebras which can be represented using triangulations of surfaces. These surfaces can have a certain number of sides but also a certain number of punctures (points inside the surfaces) and boundary components (shape inside the surface with marked points on them). For example, the triangulation at the bottom left of the poster has 6 sides, 1 puncture and 2 boundary components, one with 6 marked points on it and the other with 3 marked points.

Given a surface with a given number of sides, punctures, and boundary components, a triangulation of the surface will represent a specific cluster X (set of cluster variables) and the mutation μ of a certain cluster variable in X will have the same behavior as the "flip" of a certain diagonal in the triangulation. The "flip" of a diagonal consist of erasing one diagonal of a triangulation and replacing it with the other diagonal that creates a new valid triangulation.

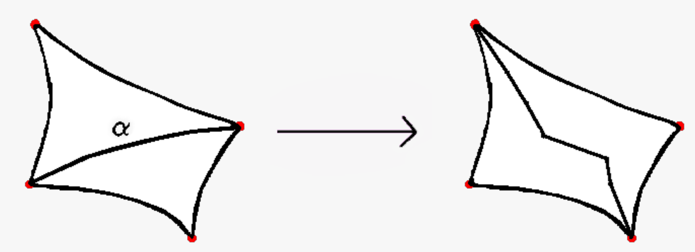
The problem I was given is the following: Given an arbitrary 2D triangulation chosen by a user of the software, and given a diagonal in this triangulation, implement an algorithm which will automatically draw the other valid diagonal which will replaced the flipped one. The new triangulation it creates corresponds to the unique valid cluster variable x_i' obtained when mutating x_i, one of its cluster variable.

Different triangulations

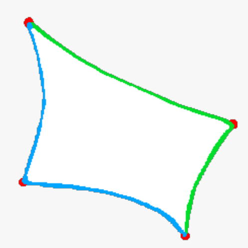


An Algorithm for Diagonal flip

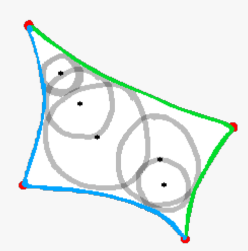
Once a triangulation has been given by the user, any diagonal in the triangulation should be able to be flipped. The problem of diagonal flip for an arbitrary triangulation can be brought back to being able to flip the diagonal of an arbitrary quadrilateral. This is an overview of the algorithm which creates the new diagonal.



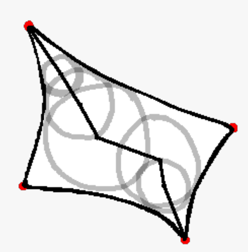
We start by identifying the sides of the quadrilateral. Two sides are the same colors if they are connected by a point that the original diagonal α started or ended at. We therefore get the following coloring for our quadrilateral.



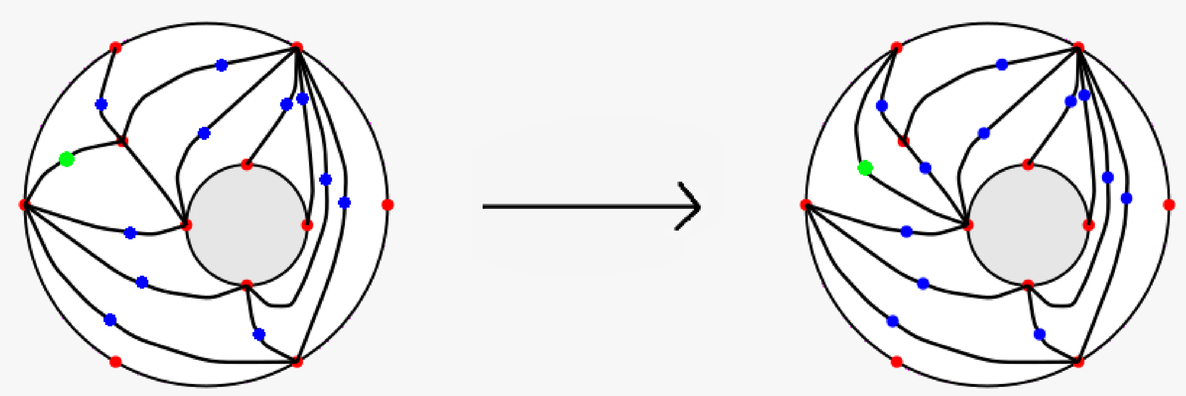
We then draw the biggest circles that fit fully within the quadrilateral and that touches both lines.



We then connect the center of each circle to form a valid diagonal contained within the parallelogram.

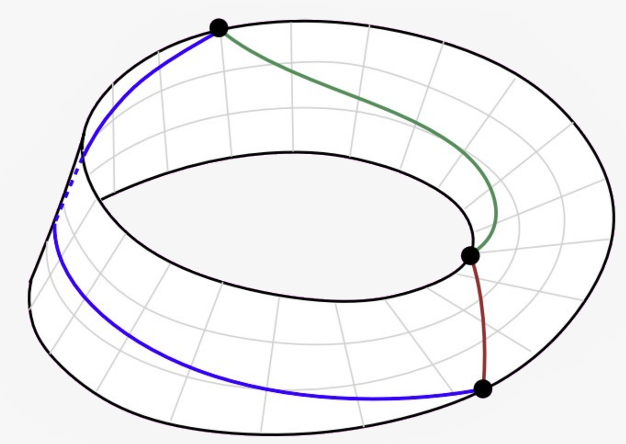


Diagonal Flip for any Triangulation



What to do next? - Computations in Cluster Algebra

Being able to draw and compute the diagonal of more complicated surfaces like non-orientable surfaces like the mobius strip[1] or surfaces in 3 dimensions would be a good addition to the software.



I would also like to compute the cluster variables the triangulations represent. This would require to add tagged arcs in the software which are fundamental in cluster algebras of triangulated surfaces.

It would also be interesting if one could compute other values related to the cluster algebra like the quiver of the triangulation, the F-polynomial, the g-vector and the c-vector. This would make the software very interesting for studying cluster algebra of surfaces. The software would also be a nice learning tool, providing a visual way to understand cluster algebra.

How to find the software?

The working version of the diagonal flip algorithm is written in Python and is difficult to share. However, I am currently working on a JavaScript version of Triangulation Software which will be available on my website.

Acknowledgments

I want to thank Véronique Bazier-Matte for having proposed this project and for supervising me. I also want to thank the FRQNT for this research grant which allowed me to have this first experience in research.

DOI: https://doi.org/10.69777/356806

References

This article from my supervisor which is an introduction to triangulation of surfaces in cluster algebra (the mobius strip image is from this paper):

[1] V. Bazier-Matte, R. Huang and H. Luo, Number of Triangulations of a Mobius Strip, https://arxiv.org/pdf/2009.05785

And this is the article in which triangulations related to cluster algebra where first introduced:

[2] S. Fomin, M. Shapiro, D. Thurston, Cluster Algebras and Triangulated Surfaces Part I: Cluster Complexes, https://arxiv.org/pdf/math/0608367