Tim Valencony
14/06/2023

CSE306 Assignment 2 Report

1. Introduction:

This project is part of the CSE306 course taught at Ecole Polytechnique in 2023. It shows the implementation of the following algorithms: Voronoï Parallel Linear Enumeration with Sutherland-Hodgman polygon clipping algorithm, an extension of this to the Power diagram, an optimization of the weights of the power diagram using LBFGS, and an attempt at the Gallouet-Mérigot incompressible Euler scheme implementation.

I could not finish this project entirely as I couldn't figure out how to make the Gallouet-Mérigot incompressible Euler scheme work, it currently only shows blank png files, but I hope I can get partial credit for what was wrongly implemented.
Finally, you will find the code for both TD5 and TD9 in the repository as well.

2. Code Structure:

The code is segmented in different TD directories, labeled from 5 to 9. In each lab, there is a "main.cpp" file, which inherits from other files such as "vector.cpp", "voronoi.cpp", … etc which implement classes needed depending on the TDs. In the "vector.cpp" file, the code was mainly reused from the previous assignment.

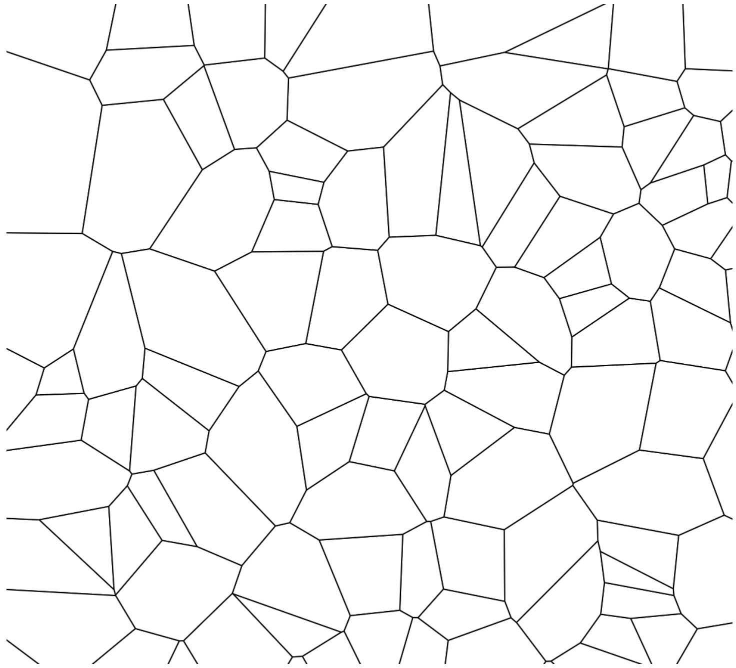
3. TD5:

Here is a quick illustration of lab 5 just to make sure it was indeed done:

4. TD6:

We implemented the Sutherland-Hodgman algorithm which allows us to clip polygons by convex polygons. We updated each line iteratively in order to build a new polygon. This allowed us to implement the Voronoi Parallel Linear Enumeration, even though I could not use omp.h (Mac with an M1 chip) so the correct line of code was commented.

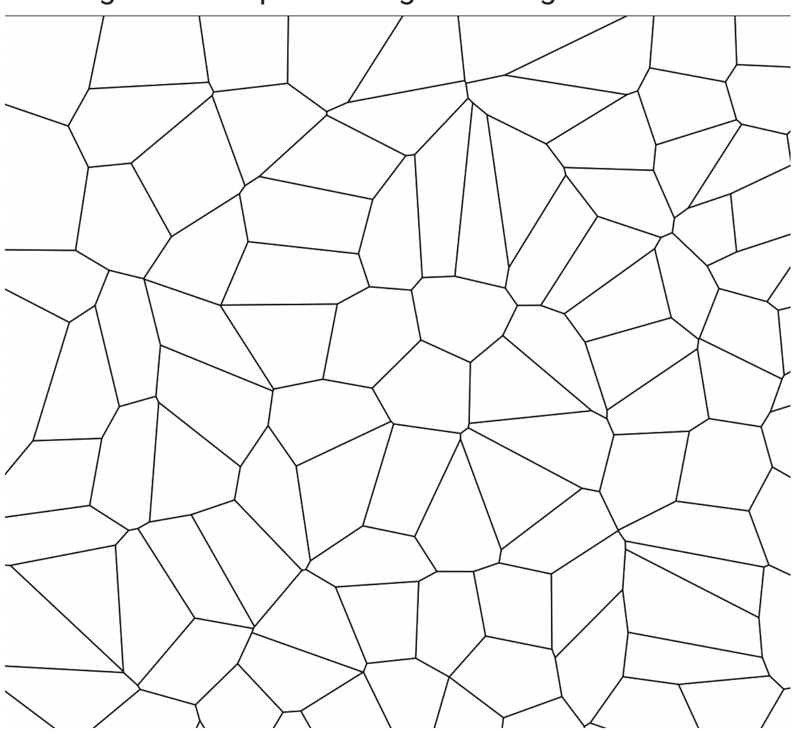
Tim Valencony
14/06/2023



Voronoi diagram for 256 points

5. **TD7:**

In lab 7, we changed the previous algorithm to incorporate the power diagram, which allows us to control the size of each cell via a set of weights. Here we set an optimization of the weights of the power diagram using LBFGS.



Power diagram for 256 random points

6. **TD8:**

Nothing to illustrate here unfortunately, if you want to check the code, I left the implementation in the github repository anyways.

7. **TD9:**

See .obj images in the github repository.

8. **Feedback on the course:**

I personally liked the course, and found it insightful into the graphics we are seeing everyday and how they have evolved. It really gave me a sense of all the physics that is behind every single component and that really amazed me (including all of the formulas and how we could implement each and everyone of them in code).

In terms of the structure of the course, I can say that for the first project it was pretty hard for me personally as I found that the help in the textbook wasn't really clear (meaning it was hard for me to find a "thread" to follow clearly and find out exactly where in the "thread" I was wrong).

However, whenever I asked a question on slack, the professor of the TA replied daily and I found that to be very professional / helpful. So kudos for that, it shows that the teaching staff of this course is very involved.

9. **Sources:**

The code was largely inspired by what was coded in class by the professor (Mr. Nicolas Bonneel). Furthermore, when having trouble in figuring out how to move forward, this and this github repositories helped me in unlocking the situation.