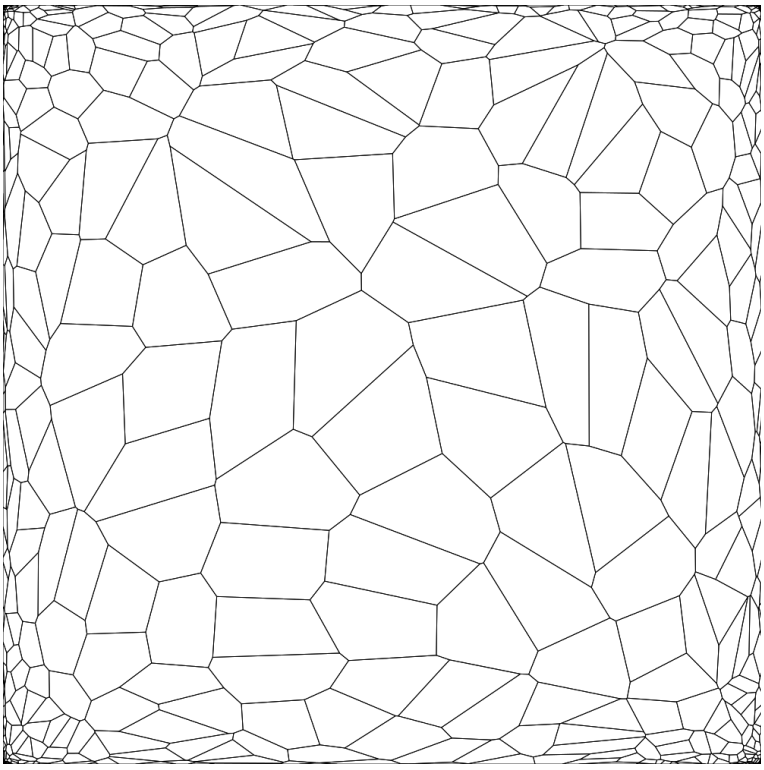
# CSE306: Geometry Processing Project Report

Joshua P. Jacob (BX21)



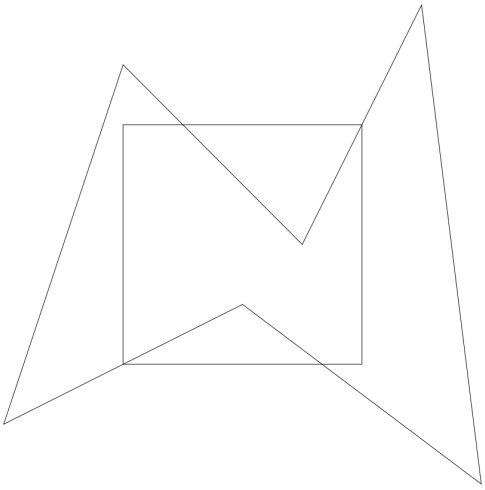
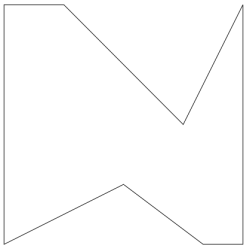
## Introduction

In this project, the following algorithms/schemes were implemented: Sutherland-Hodgman algorithm for polygon clipping, 2D Voronoï/power diagram generation using parallel linear enumeration, and optimal transport for power diagram  weight optimization with L-BFGS. Unfortunately, I did not succeed with 2D fluid dynamics but my attempt of the Gallouet-Mérigot Incompressible Euler scheme is still in main.cpp.

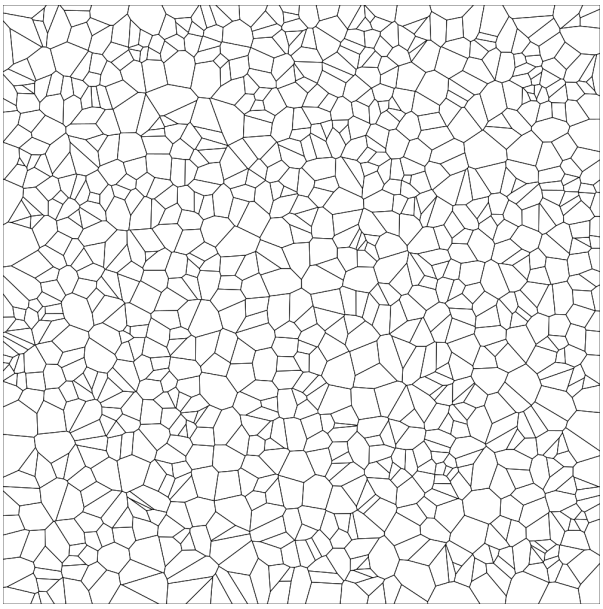
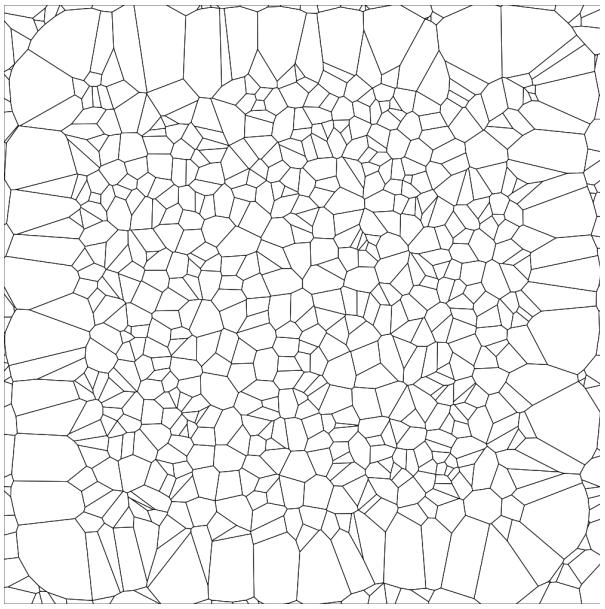
## Code Structure

The code is organized as follows. The implementation of the Vector class is in vector.cpp (same as in previous projects). Functions to create svgs and animation frames are in svg.cpp along with the Polygon class. Main.cpp contains all the code for polygon clipping, voronoi/power diagram generation, tutte embedding, and fluid dynamics (although broken). Lbfgs.cpp contains the code for power diagram weight optimization using liblbfgs.

## Polygon Clipping

| Before Clipping | After Clipping |
| --- | --- |
|  |  |

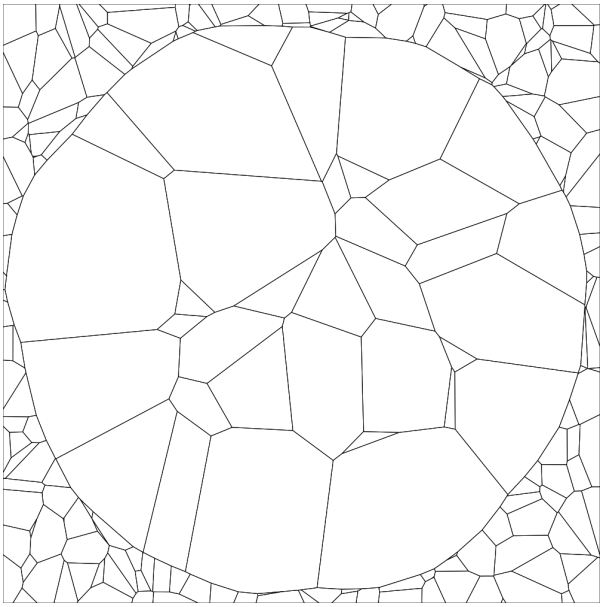
## Voronoï/Power Diagrams

| Voronoï - naïve O(N^2) parallel linear enumeration | Power (code for this example is in main.cpp) |
| --- | --- |
|  |  |

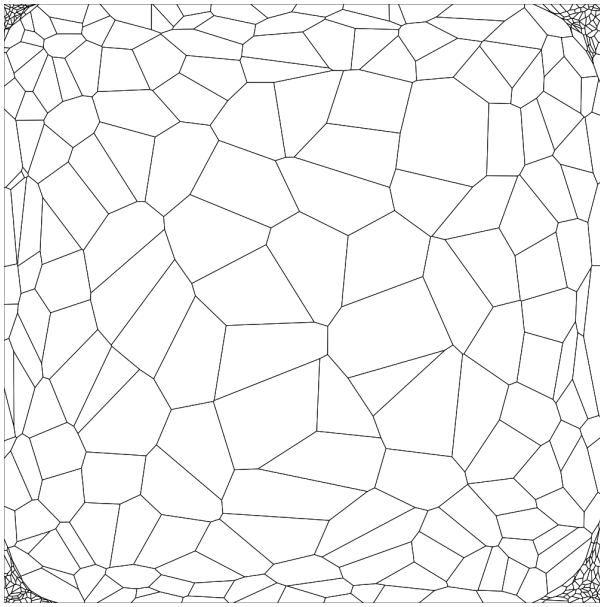
# Power Diagram Weight Optimization

The following example was optimized using semi-discrete optimal transport similar to the example in the notes (centered Gaussian). 1000 points were distributed uniformly initially.

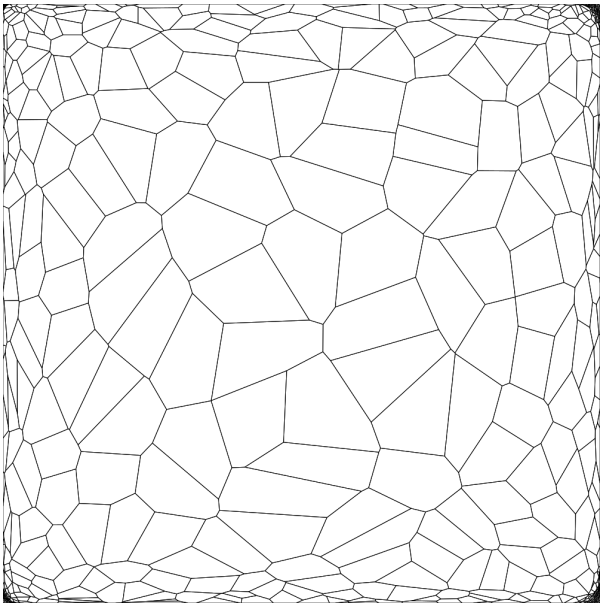
After 10 Iterations,



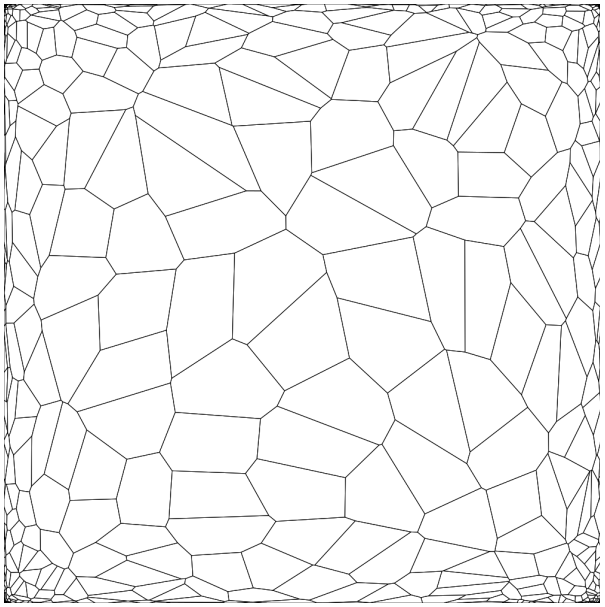
After 100 Iterations,



After 500 Iterations,



Final (~1300 Iterations),



# Computational Fluid Dynamics

I did not succeed completely with this. My attempt is still in main.cpp. I hope I can get some partial credit.

## More

Furthermore, Lab 5 sliced optimal transport color matching and tutte embedding were also implemented. Tutte embedding is in main.cpp and color matching is in a different folder (separate from the rest).