

Report for Advance Algorithm Programming Assignment 1

Your name

1 Implementation Details

To implement Delaunay triangulation in 3-dimension, I used 'QHull()' to compute convex hull of the vertices in 4-dimension.

Given set of vertices $V = \{(v_x, v_y, v_z) \in \mathbb{R}^3 \mid \text{for } v_x, v_y, v_z \in \mathbb{R}\}$, we lift up the points $v \in V$ by adding additional component $V_m = (v_x)^2 + (v_y)^2 + (v_z)^2$. Let us denote $V4 = (v_x, v_y, v_z, v_m) \in \mathbb{R}^4$. Then we can construct a convex hull of $V4$ which is a composition of tetrahedrons. After computing the convex hull of $V4$, we select the vertices of the tetrahedron whose facing direction is downside. After storing the information of that selected vertices $v_{selected} \in V4$, we draw tetrahedron of $v_{selected}$ with

2 Example Output

3 Known bugs/limitations

References