INFODDM Exercise 1 - Mesh Representations

Question 1

You are asked to model a cube in mesh geometry on paper. The cube has sides of length 1 and its lower left corner sits at the origin of the Cartesian coordinate system.

(a) Sketch the cube (preferably on graph paper), assign a letter to each corner (vertex) of the cube. Then write in a separate list the letter of each corner and the corresponding coordinates of that vertex.

(b) If we are not interested in the spatial layout of this cube we could represent it as a (flattened) graph. Draw the graph of this cube, make sure to indicate the letters of each vertex at the correct location.

What problem occurs when you try to draw this cube as a graph in the plane? Do you see a way to work around it?

(c) The graph representation of the cube is the first step of building a mesh for the cube. But an essential element is still missing, we need to consider the faces of the cube as well. When we consider the planar graph to include faces, would you classify this graph as a *triangular*, *quad* or *semi-regular*?

(d) The form of the cube can be represented really well by a *winged edge structure*, though tedious to write down it will capture all of the geometric and connectivity information of the cube. As such it can easily be converted into the other mesh representations.

Write the three lists that form the winged edge structure of the cube. Use CCW ordering. Try to be as compact as possible (consider optional elements), while choosing short but representative names for each of the winged edge structure elements. Make sure you use the right terms and be consistent when writing down the information.

(e) Choose an arbitrary vertex of the cube, what is the degree of this vertex? Is there a way to infer this easily from the winged edge representation? Is there an ordering that you need to be aware of?

(f) You will have noticed that the degree of each vertex is the same, this **particular degree** implies that the mesh is *manifold*. Explain why this is the case.

(g) Calculate the genus of the cube using the Euler characteristic. Could you have gauged the genus from your initial sketch? How?

Question 2

We are still considering the cube from question 1. Unfortunately, many applications require meshes to be triangular, in particular when meshes are drawn on screen. This is why we will have to triangulate the cube.

(a) What is the reason that triangular meshes can be drawn easily while polygonal meshes cannot?

(b) Count the number of ways the cube can be triangulated.

(c) Choose a triangulation you like, then write down the *triangle list* of this mesh triangulation.

(d) Without calculating the Euler characteristic explicitly, deduce from the triangulation operation itself how the Euler characteristic changes. In particular notice how the number of edges and faces change.

(e) Write down the DCEL of the chosen triangulation. Make sure you use the correct notation with the least amount of redundancy.

(f) Having the DCEL of a mesh makes certain queries on the mesh very simple, can you give an example of a query that is easily evaluated using the DCEL?

(g) You will notice that the triangle list does not contain any edge or face connectivity information, there is no way to see whether each triangle is truly connected to any other. They might as well be free floating triangles. Creating such a free floating mesh from another is often called an *explosion* or an exploded mesh. How would you classify exploded meshes in the Representation Zoo.

Question 3

Very often, just having the geometric information of a mesh is not sufficient to solve certain modeling problems. We often want to add additional data to the mesh. When we do this we say the mesh is *augmented*. We want to include the following information in the mesh, the length of each edge, the name of the vertex, and whether a face is oriented up, left, right and so on.

(a) Propose a few methods to include this augmented data into the following mesh representations: DCEL, winged edge mesh and triangle list.

(b) Consider the cube from assignment 1, if we stack many cubes together on a grid we could also associate augmented data with each cube on that grid (assuming there is a cube in that grid cell). This effectively adds an extra layer of augmented data to the geometry. How do we call such as system?

(c) Can you propose a better representation that uses the fact that each cube is the same, that uses much less data, while keeping all the augmented data and the triangular structure of the cubes?

(d) Finally, we want to calculate the distance to the closest cube in the system. Describe, informally, an algorithm that calculates this distance using the fact that there are only cubes in the system.

(e) This distance algorithm, when seen as a function is also a form of geometry. How do we call this geometry?