1 Soccer Ball Triangulation

1.1 Triangle Data

The file `triangulation.mat` contains two arrays, `htria` and `ptria`, of lengths 1080 and 540. These arrays encode x, y, z coordinates of vertices of triangles corresponding to divisions of hexagons and pentagons as described in the Plots section below. The data for a triangle with vertices A, B, C is provided in the following order: `xA, yA, zA, xB, yB, zB, xC, yC, zC`. The data for triangles follow each other, i.e. you can read the coordinates using for loops.

You can load the data inside a `.mat` file into your workspace by calling the `load` function inside your script:

```
load('triangulation.mat');
```

1.2 Random Ray Generator

The function `rand_isotropic` returns a random ray hitting the surface of a sphere of radius 1. The output is x,y,z coordinates corresponding to a direction.

1.3 Ray-Triangle Intersection

If you didn't write the ray-triangle intersection function, you can make use of the following function which is available from Matlab central.

When you extract the zip file, *TriangleRayIntersection.zip*, you will notice a function called `TriangleRayIntersection`. You can call this function as follows:

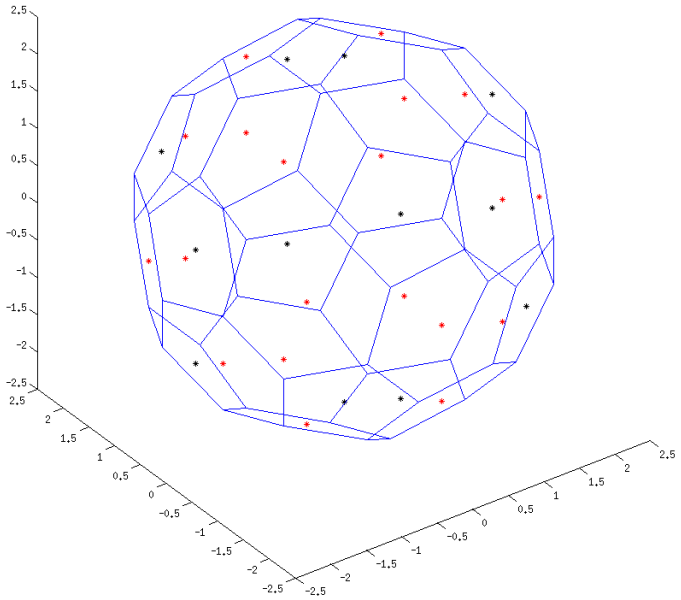
```
[y, t] = TriangleRayIntersection(orig, dir, vA, vB, vC);
```

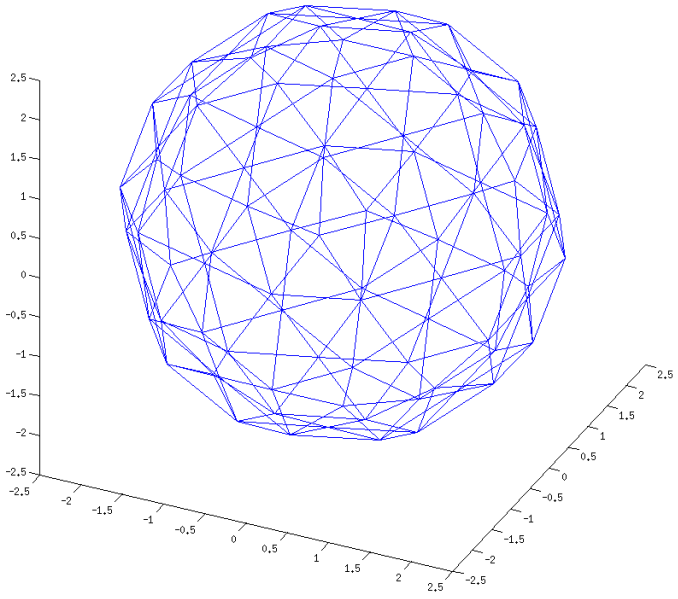
- `orig` is the coordinates for the origin, i.e. `[0, 0, 0]`
- `dir` is the direction of random ray
- `vA,vB,vC` are the coordinates of vertices A,B,C
- If `t>0` and `y` is 1, then the ray intersects the triangle.

This function can also be used in many other ways which are listed in the helper comments.

2 Plots

This is just to give you information about the triangulation data provided. The hexagon and pentagon centers on a truncated icosahedron are shown using red and black dots respectively. Each polygon is divided into 5 or 6 triangles where one of the vertices is the center point and the other two are neighbor vertices of the polygon.


2.5
2
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−2.5


2.5
2
1.5
1
0.5
0
−0.5
−1
−1.5
−2
−2.5