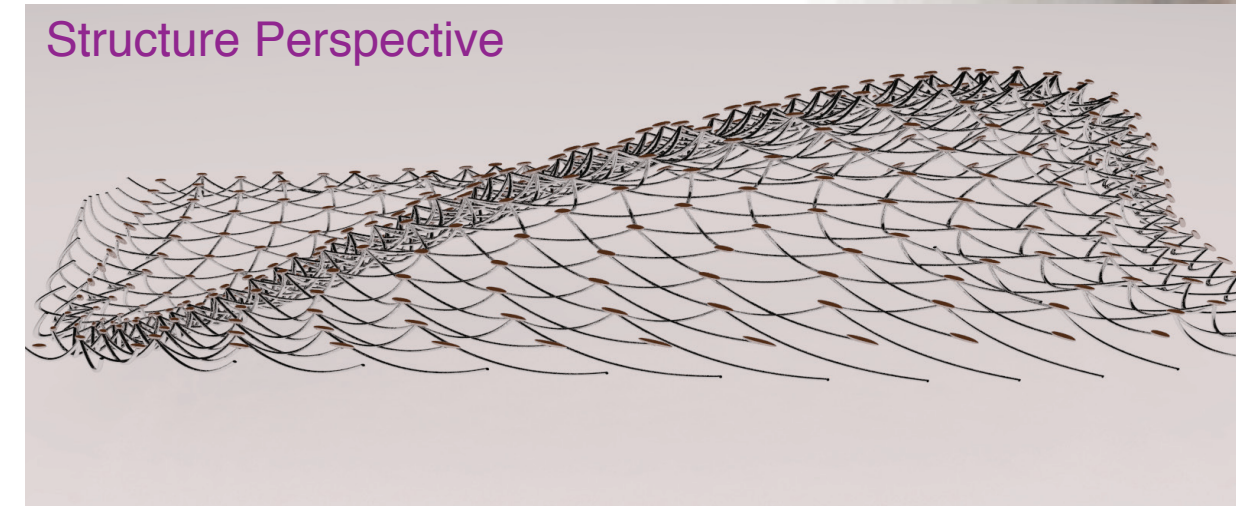
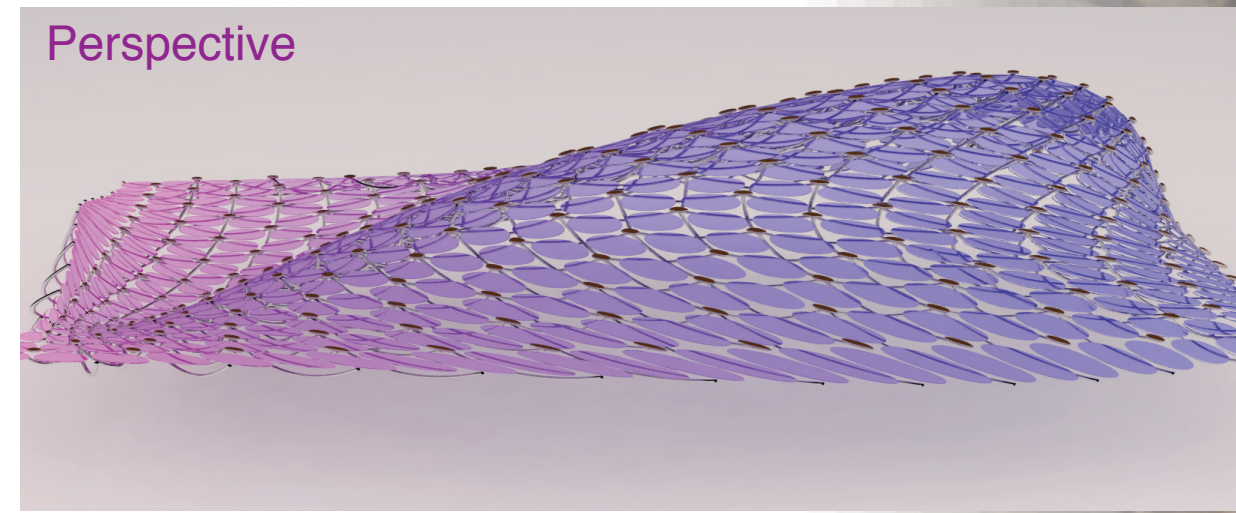
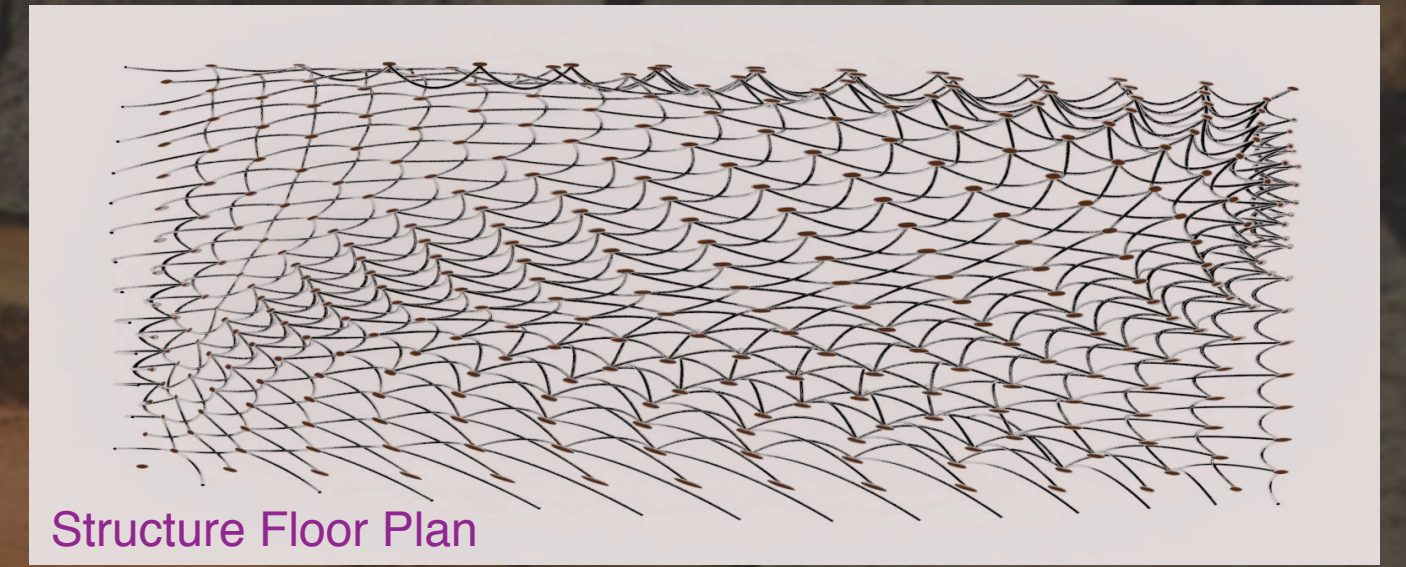
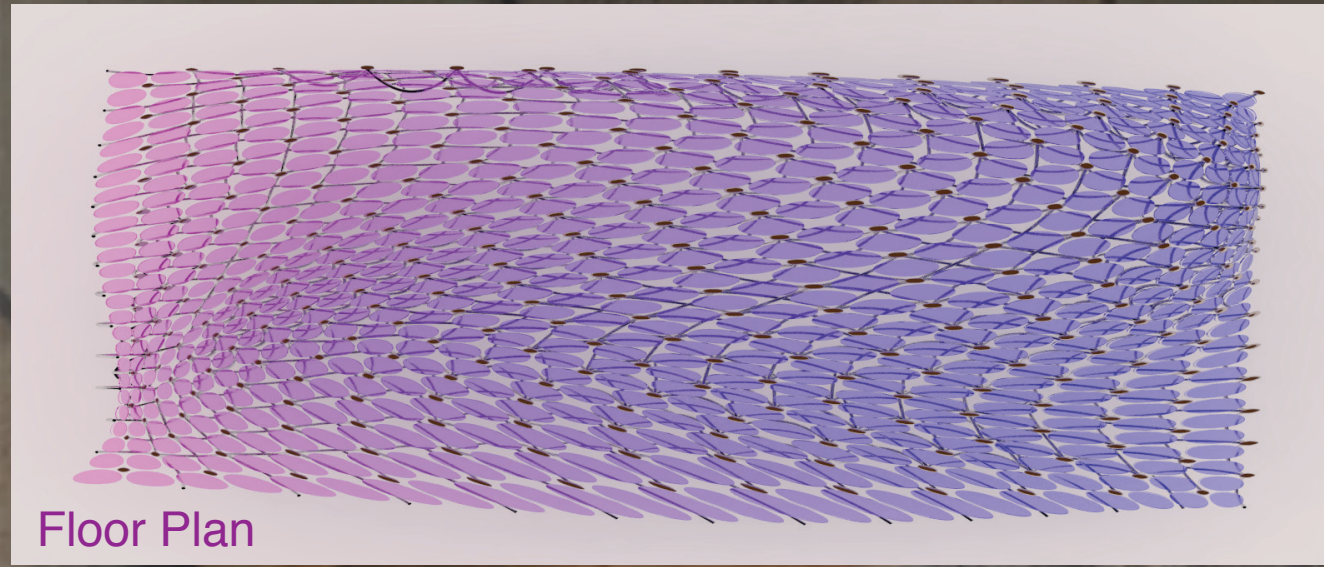


Circular Panels

The use of an Euclidean system of coordinates applied to a given surface, allows the application of three-dimensional panels, controlled by normal vectors, on a surface.

Using second-degree curves it is possible to create circular panels as well as structure and spherical-like supports.



```

sketch_oct19a | Processing 1.5.1
STANDARD
sketch_oct19a
import processing.opengl.*;

import igeo.p.*;
import igeo.io.*;
import igeo.geo.*;
import igeo.util.*;
import igeo.core.*;
import igeo.gui.*;

size(1600,900,IG.GL);

IG.open("surface5.3dm");

ISurface[] surfs = IG.surfaces();
IImageMap map = new IImageMap("grad.jpg");

for( ISurface surf : surfs ){

  ILayer layer1 = IG.layer("joints");
  ILayer layer2 = IG.layer("structure");
  ILayer layer3 = IG.layer("panelees");

  int unum = 25, vnum = 25;
  double uinc = 1.0/unum, vinc = 1.0/vnum;
  double offsetDist = -0.04;
  int degree = 3;
  double z = 0;
  double sc = 2;

  for(int i=0; i<=unum; i++){
    for(int j=0; j<=vnum; j++){
      double val = map.get(i*uinc,j*vinc);

      //Large Circle
      IVec pta = surf.pt ( i*uinc, j*vinc,z);
      IVec ptb = surf.pt ( (i+1)*uinc, j*vinc,z);
      IVec ptc = surf.pt ( (i+1)*uinc, (j+1)*vinc,z);
      IVec ptd = surf.pt ( i*uinc, (j+1)*vinc,z);

      //Small Circle
      IVec ptas = surf.pt ( i*uinc, (j-sc)*vinc,z);
      IVec ptbs = surf.pt ( (i+sc)*uinc, j*vinc,z);
      IVec ptcs = surf.pt ( i*uinc, (j+sc)*vinc,z);
      IVec ptds = surf.pt ( (i-sc)*uinc, j*vinc,z);

      //Structure behind
      IVec str1 = pta.dup();
      IVec str2 = surf.pt ((i+1)*uinc,j*vinc,-2);
      IVec str3 = surf.pt ((i+2)*uinc,j*vinc);
      IVec str4 = surf.pt (i*uinc,(j+1)*vinc,-2);
      IVec str5 = surf.pt (i*uinc,(j+2)*vinc);

      ICurve Lcrv = new ICurve (new IVec[] {pta,ptb,ptc,ptd},degree,true);
      ISurface panel = new ISurface(Lcrv).clr(i*uinc,0,9).layer(layer3);

      IG.flatten(Lcrv);
      IG.extrude(Lcrv, -05).clr(i*uinc,0,9).layer(layer3);

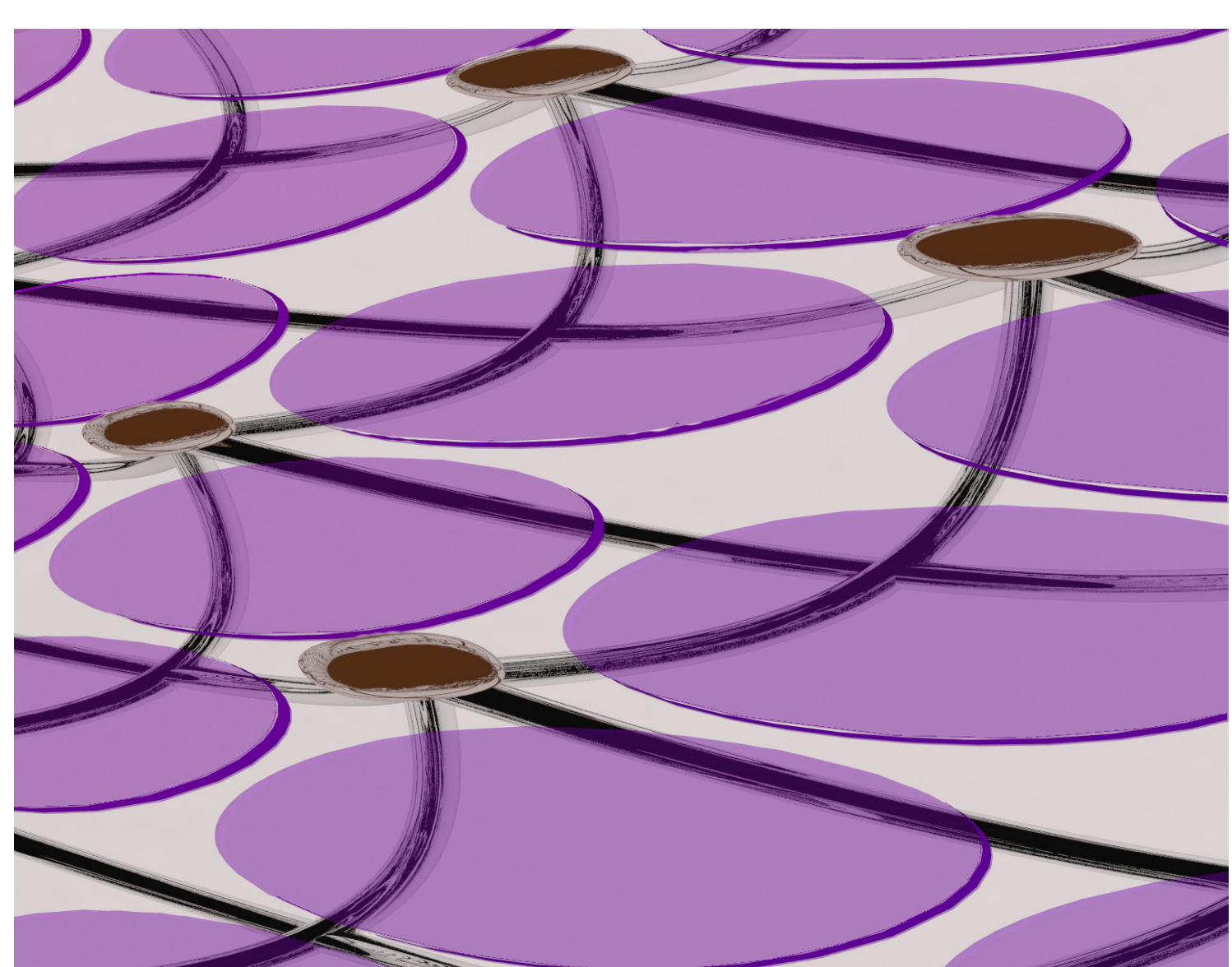
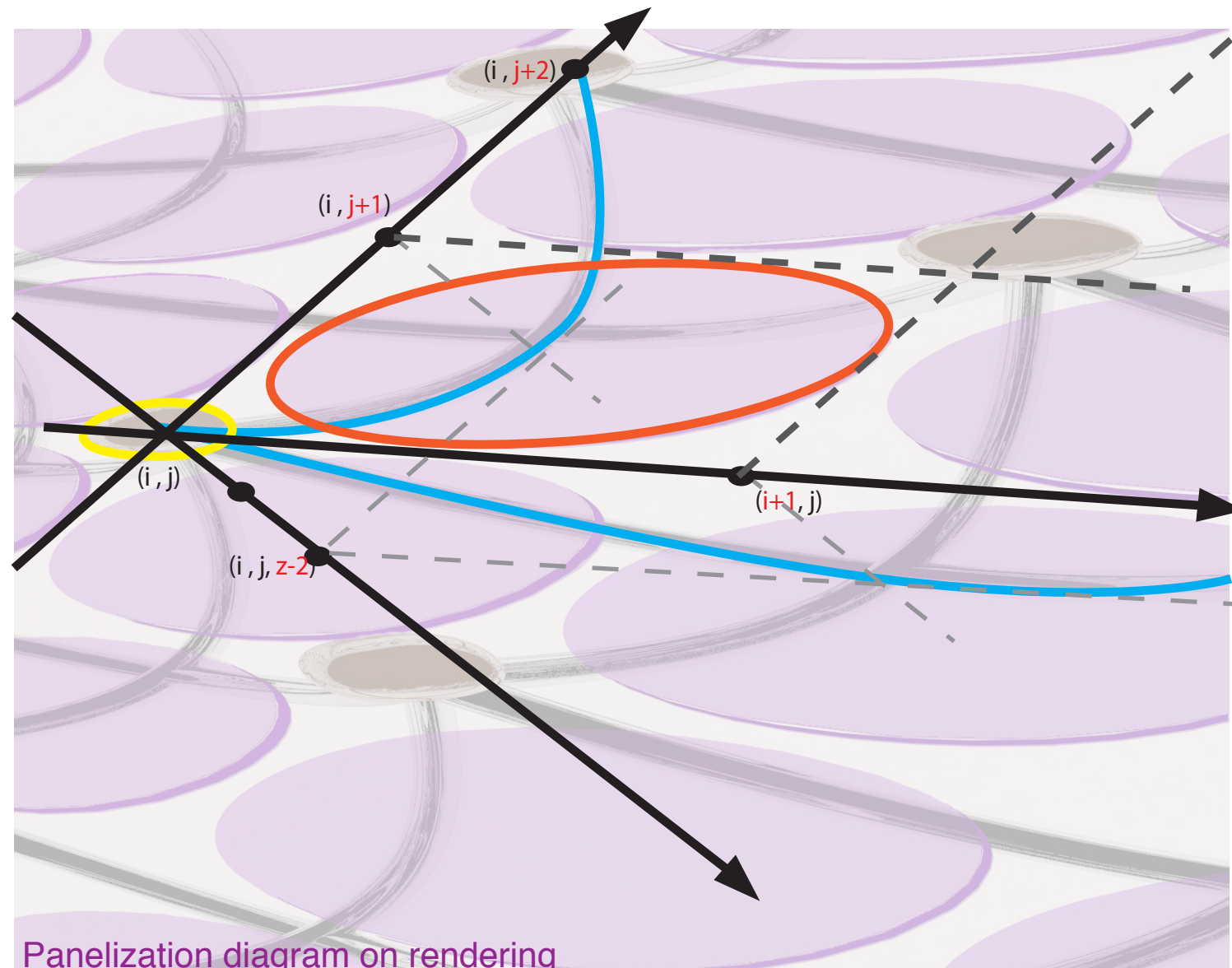
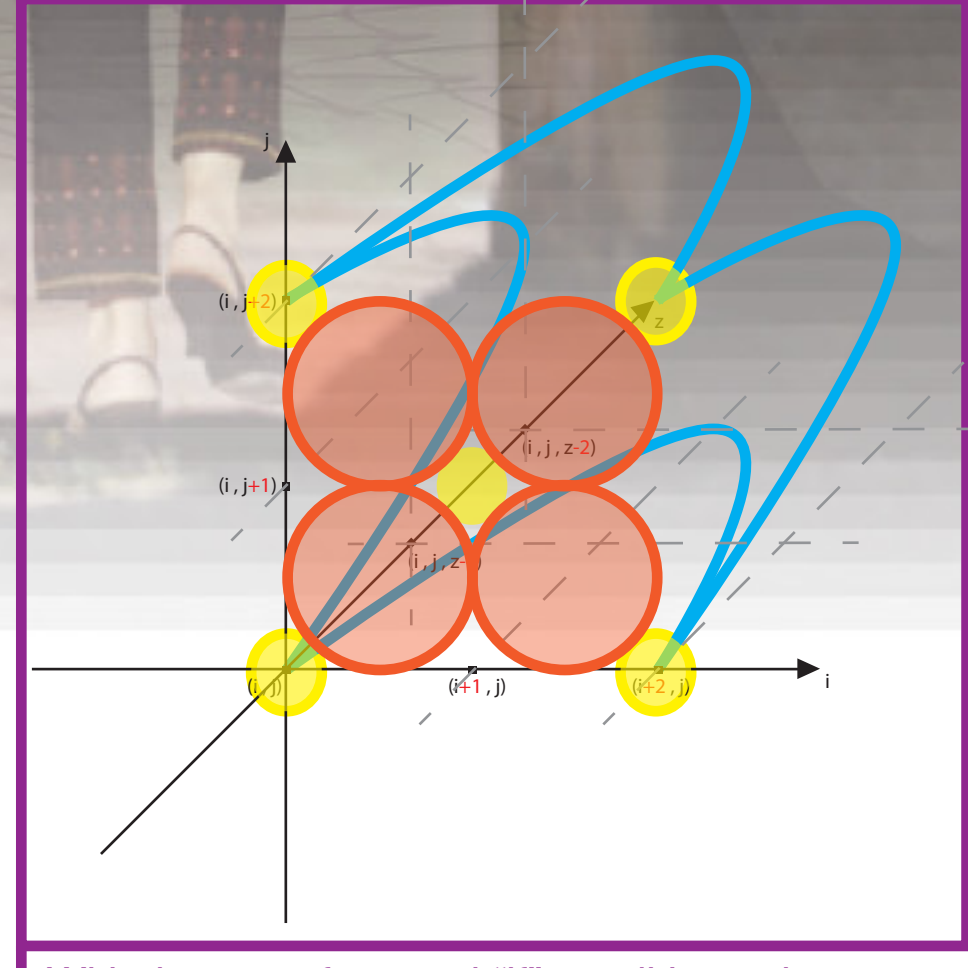
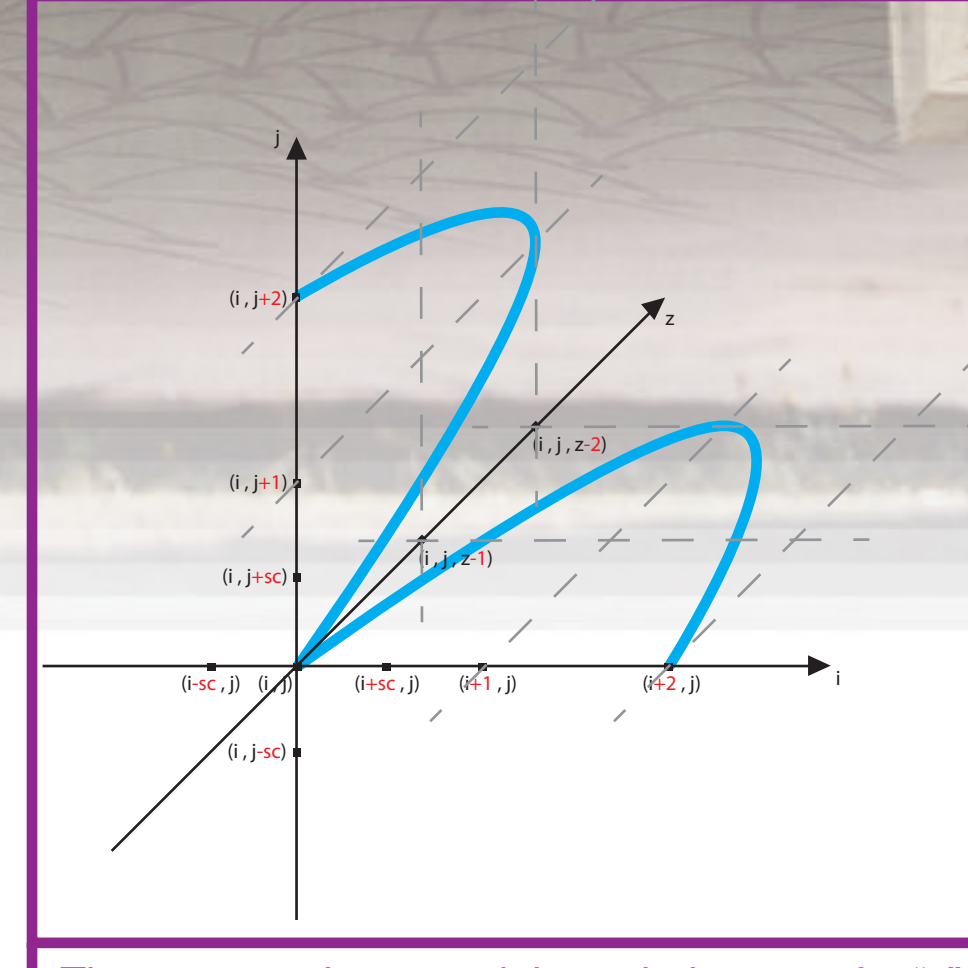
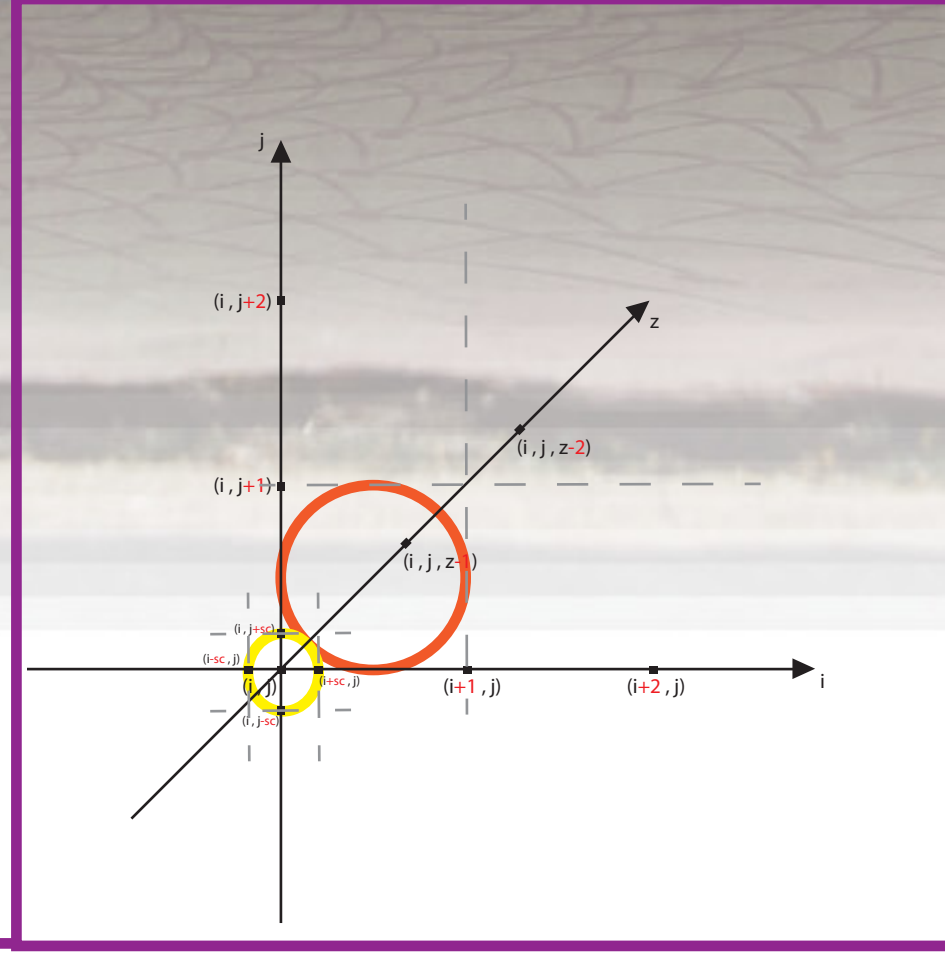
      if( (i+j) % 2 == 0){
        if( i < 25 ){
          if( j < 25 ){
            ICurve structure1=new ICurve (new IVec[] {str1,str2,str3,2});
            ICurve structure2=new ICurve (new IVec[] {str1,str4,str5,2});
            IG.pipe(structure1, (sc*.3)).clr(0,1,0).layer(layer2);
            IG.pipe(structure2, (sc*.3)).clr(0,1,0).layer(layer2);
          }
        }
        double blobDepth = 0.2;
        IVec ptas2 = surf.pt ( i*uinc, (j-sc)*vinc,z+blobDepth);
        IVec ptbs2 = surf.pt ( (i+sc)*uinc, j*vinc,z+blobDepth);
        IVec ptcs2 = surf.pt ( i*uinc, (j+sc)*vinc,z+blobDepth);
        IVec ptds2 = surf.pt ( (i-sc)*uinc, j*vinc,z+blobDepth);

        IVec ptsCenter = surf.pt ( i*uinc, j*vinc,z);
        IVec ptsCenter2 = surf.pt ( i*uinc, j*vinc,z+blobDepth);

        IVec[] blobCpts = new IVec[4][4];
        blobCpts[0][0] = ptsCenter;
        blobCpts[0][1] = ptsCenter;
        blobCpts[0][2] = ptsCenter;
        blobCpts[0][3] = ptsCenter;
        blobCpts[1][0] = ptas;
        blobCpts[1][1] = ptbs;
        blobCpts[1][2] = ptcs;
        blobCpts[1][3] = ptds;
        blobCpts[2][0] = ptas2;
        blobCpts[2][1] = ptbs2;
        blobCpts[2][2] = ptcs2;
        blobCpts[2][3] = ptds2;
        blobCpts[3][0] = ptsCenter2;
        blobCpts[3][1] = ptsCenter2;
        blobCpts[3][2] = ptsCenter2;
        blobCpts[3][3] = ptsCenter2;

        new ISurface(blobCpts, 2, 2, false, true).clr(i*uinc,0,9).layer(layer1);
      }
      surf.del();
    }
  }

  IG.save("surface2out.3dm");
  
```



Student: Salvador Cortez
Tutor: Satoru Sugihara
Coding Form Midterm Fall 2011