
Subject: Re: 3d polygon mesh for 3 independent variables, x,y,z

Posted by [Dick Jackson](#) on Mon, 23 Mar 2015 20:04:25 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Thursday, 19 March 2015 03:23:54 UTC-7, guni wrote:

> Hi,

> I am a new in IDL. And I need some help.

> I have 3 independent variables. I plotted 3d polygon in MATLAB using 'convhull' and then 'trimesh' procedures. But I have to do in IDL. I already plotted a 3d scatter plot using 'plot3d' and then I go for 'qhull' for the delaunay triangulation. However, I am not able to plot the (bound) polygon mesh in my 3d plot. Or, is this triangulation not the way I should look for?

> If anyone can help how to plot such a 3d-polygon, that will be a nice pleasure.

Hi Guni,

This sounded familiar to me, and I see I wrote something like this a few years ago, before Function Graphics (search the archive for QHullTetra).

I'm not certain exactly what you are looking for as a final result. Here is how you can add the convex hull and the Delaunay triangulation (actually "tetrahedralization" in 3-D), which is a little trickier:

PRO QHullTetra2

```
;3 independent variables
```

```
nPts = 10
```

```
x=randomu(seed,nPts)
```

```
y=randomu(seed,nPts)
```

```
z=randomu(seed,nPts)
```

```
xyz = [Transpose(x), Transpose(y), Transpose(z)]
```

```
;3d scatter plot
```

```
pch = PLOT3D(x, y, z, 'o' ,/SYM_FILLED,AXIS_STYLE=2,/PERSPECTIVE, $  
    Aspect_Ratio=1, Aspect_Z=1, WINDOW_TITLE='Convex Hull')
```

```
; Convex Hull:
```

```
qhull,x,y,z,triVerts
```

```
convexHull = Polygon(x, y, z, /Data, $
```

```
    Connectivity=[Replicate(3, [1, N_Elements(triVerts)/3]), triVerts], $
```

```
    Target=pch, Transparency=80)
```

```
;3d scatter plot
```

```
pdt = PLOT3D(x, y, z, 'o' ,/SYM_FILLED,AXIS_STYLE=2,/PERSPECTIVE, $  
    Aspect_Ratio=1, Aspect_Z=1, WINDOW_TITLE='Delaunay Tetrahedralization')
```

```
; Delaunay tetrahedralization:
```

```
qhull,xyz,tetraVerts,/delaunay
```

```
nTetra = N_Elements(tetraVerts)/4
```

```
; Make four triangles from each tetrahedron
tetraConn = tetraVerts[[0, 1, 2, 0, 2, 3, 0, 3, 1, 1, 3, 2], *]
tetraConn = [Replicate(3, [1, nTetra*4]), Reform(tetraConn, [3, nTetra*4])]
delaunayTri = Polygon(x, y, z, /Data, $
  Fill_Background=0, Linestyle=0, $
  Connectivity=tetraConn, Target=pdt)
```

END

The Voronoi diagram is much trickier (see QHULL docs for VVERTICES in particular), and I don't have a final answer for that. You would start with this call to get all the data:

```
qhull,x,y,z,triangle,/delaunay,VDIAGRAM=vdiagram,$
  VVERTICES=vvertices,connectivity=connectivity
```

Then much work remains!

Hope this is helpful.
