

---

Subject: Re: Using POLYGON to plot unstructured mesh  
Posted by [Dick Jackson](#) on Fri, 25 Oct 2013 21:36:31 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Laurent Testut wrote, On 2013-10-23, 4:54am:

> Hi Dick, Thanks for your reply, specially because the POLYGON is obviously  
> not a sexy subject in this newsgroup :). You clearly understood my needs.  
>  
> My mesh is a 2D mesh from ADCIRC model (which I was able to visualize )  
>> oModel = OBJ\_NEW('IDLgrModel') oPolygon = OBJ\_NEW('IDLgrPolygon',  
>> vertices, POLYGONS = connectivity, STYLE = 1) oPolygon->SetProperty,  
>> STYLE=1, thick=1 oModel->Add, oPolygon XOBJVIEW, oModel, /BLOCK, TITLE =  
>> 'Original Mesh'  
>  
> I want now to be able to plot some properties of my model (bathymetry,  
> rugosity, velocity, etc) at each grid node or some properties associate with  
> each triangle (surface, ...). I also want to be able to simply visualize the  
> boundaries with a different color and to plot the polygon segments with a  
> size (km) dependant color scale (each segments has color with depend on his  
> length (in km). As you can see I want to do a lot of things. I was wondering  
> if there is a simple way to associate to my polygon (x,y) and connectivity  
> some additional 'properties/data' (bathy, velocity, ...) which can be easily  
> added to my polygon objects and then plotted.  
>  
> Thanks for your help, Laurent

Great, thanks for the detail. You're looking for full control of colours of polygon fills, segments and perhaps vertices as well.

The IDLgrPolygon class is an efficient way to work with such meshes if you don't need more flexibility than it offers.

Using one IDLgrPolygon object, created in the usual way (with vertices shared between triangles) and its Vert\_Colors property, you can indeed control the colour of the polygon fills (Style=2) or the colour of the segments (Style=1), or by using two of them you could have segments overlaid on the filled mesh using different sets of colours. Why stop there? A third object with Style=0 could have yet another set of colours used to make points at each vertex! When you want a different dataset represented, you just use  
object -> SetProperty, Vert\_Colors=newValues  
and the next time it is rendered, the new colours are used.

However, for the polygon fills, "The color of the first vertex in each polygon is used to define the color for the entire polygon." This is a problem for your task, as there can be more triangles than vertices, and it would be difficult to coordinate this for perfect triangle-colour control. For segment colours, there would be similar issues.

Now, it's possible, although a little less efficient, to create the mesh with three new vertices for each triangle, and control of the fill colour is then easy. Similarly, the polygon segments could be duplicated in an IDLgrPolyline object, also with Vert\_Colors, where, "the color of a line segment is the color of the second vertex of that line segment."

(Another possibility is to work with a texture map image created to show your data, but I'm not sure that is quite what you're looking for.)

It seems that the Function Graphics POLYGON and POLYLINE have the same sets of features as the IDLgr\* classes, so it's possible to go that way as well. I think a well-constructed object class, with these properties associated with its polygons and its mesh segments, and a widget program to allow control of display of these features, would be a very feasible project in IDL.

One more thing: You've spoken of this as a 2-D mesh, but of course, with bathymetry data added as the Z values, you will get a 3-D surface rendering of the ocean floor!

Lots of ideas, and I'd be happy to discuss this further.

--

Cheers,  
-Dick

Dick Jackson Software Consulting  
Victoria, BC, Canada  
[www.d-jackson.com](http://www.d-jackson.com)

---