Subject: Re: Create DXF file out of vertices and connectivities
Posted by Karl Schultz on Tue, 29 Jun 2004 21:26:33 GMT
View Forum Message <> Reply to Message

"Dick Jackson" <dick@d-jackson.com> wrote in message
news:B78Ec.933959$oR5.650290@pd7tw3no...
> [Please ignore my previous post, I improved my testing code for clarity]
>
> "Karl Schultz" <kschultz_no_spam@rsinc.com> wrote in message
> news:<10e0jrc69hn5715@corp.supernews.com>...
>>
>> "Tukee10" <turgutkaracay@hotmail.com> wrote in message
>>
>
news:e6feeb9d9fa4d9dad40235e15f6ab40b@localhost.talkaboutpro gramming.com...
>>> Hello,
>>> I have following problem: I created the mesh structure of a volume
> with
>>> INTERVAL_VOLUME, which gives me vertices (3,n) and connectivities
> (1,m).
>>> When I display it with XOBJVIEW, I get a wonderful 3d structure. How
> do I
>>> create a DXF file that contains the mesh structure ?
>>> I know that it works with an IDLffDXF object.
> [...]
>>> It creates a DXF file, but when I open the file with AutoCAD, it
> seems to
>>> connect all the point in a strange way.
>>> Is there anyone who has an idea ?
>>
>> INTERVAL_VOLUME returns a list of vertices and a connectivity list
> that
>> represents tetrahedra (3D pyramid-like objects). If you used this
> vertex
>> list and this connectivity list to create an IDLgrPolygon that you
> then
>> displayed in XOBJVIEW, you should have seen quite a mess - or you got
> very
>> lucky.
>>
>> You should use TETRA_SURFACE [...]
>
> "Tukee10" <turgutkaracay@hotmail.com> wrote in message
>
news:2d0234904ddaf48a12f5006bb0ae4bdb@localhost.talkaboutpro gramming.com...
>> Thank you for your response.
>> I forgot to add, that I`m using TETRA_SURFACE to display in XOBJVIEW,
>> which worked quite well. It displays the mesh structure I want, but

> when I
>> write a DXF file, the mesh looks weird when I open it with AutoCAD. Is
>> IDL_DXF_POLYGON the right structure format, if yes, does the DXF_TYPE
> has
>> to be 10 and are there other parameters to consider ?
>
> Hi Tukee, Karl,
>
> I tried this out too, and indeed it seems the file is mucked up. It does
> look like a DXF writing bug to me. If you want to look further, here's
> what I did:
>
> =====
>
> PRO WriteDXFTest
>
> ;; Make a surface (from online help for Interval_Volume)
>
> RESTORE, FILEPATH('clouds3d.dat', $
> SUBDIRECTORY=['examples','data'])
> INTERVAL_VOLUME, rain, 0.1, 0.6, verts, conn
> conn = TETRA_SURFACE(verts, conn)
> ; oRain = OBJ_NEW('IDLgrPolygon', verts, POLYGONS=conn, $
> ; COLOR=[255,255,255], SHADING=1)
> ; XOBJVIEW, oRain, BACKGROUND=[150,200,255]
>
> ;; Simplify mesh
>
> ntri=mesh_decimate(verts,conn,connout,vertices=vertsdec, $
> percent_poly=10)
> verts=vertsdec
> conn=connout
>
> polyColor = [255, 0, 0]
> dxfFile = 'Mesh_Model.dxf'
> XObjView, Obj_New('IDLgrPolygon', verts, Polygons=conn, $
> Color=polyColor), $
> Title='IDLgrPolygon Made From Raw Data'
>
> ;; From Tukee10
>
> ptr_verts = ptr_new(Verts)
> ptr_conn = ptr_new(Conn)
>
> oDXF = OBJ_NEW('IDLffDXF')
> poly = {IDL_DXF_POLYGON}
> poly.vertices = ptr_verts
> poly.connectivity = ptr_conn

```
>  poly.dxf_type = 10
>
>  oDXF->PutEntity, poly
>  IF (not oDXF->Write(dxfFile)) THEN PRINT, 'Write Failed.'
>  OBJ_DESTROY, oDXF
>
>  ;;    Try it out
>
>  oModel = Get_DXF_Objects(dxfFile)
>  oDXFRead = oModel -> Get(/All)          ; IDLgrPolygon
>  oModel -> Remove, /All
>
>  ;;    Vert_Colors are all black, change color to
>  ;;    something more visible
>  oDXFRead -> SetProperty, Vert_Colors=-1, Color=polyColor
>  XObjView, oDXFRead, Title='Read From DXF File'
>
>  END ; WriteDXFTest
>
>  =====
>
>  ... it looks like mostly-correct points are being joined up in a
>  haphazard way, with a lot of joining of points in sequence along x and
>  y.
>
>  Are we missing something in the DXF writing?
>
```

Well, yes and no.

A dxf_type code of 10 is DXF FACE3D, which is a 4-vertex polygon. I think that IDL will still try to write FACE3D entities if you use this code even though your data isn't arranged this way. This will lead to incorrect, or at least, unintended output in your dxf file. The DXF heart example that ships with IDL contains FACE3D entities, and this might lead some to try to use this code to write entities. I think that if you had a bunch of 4-vertex polygons, a code of 10 would work.

You can try a code of 9 (polygon). Yes, I know that the IDL docs don't say that 9 is valid for a IDL_DXF_POLYGON structure, but it is. IDL actually ends up generating DXF POLYLINE entities with the appropriate settings to indicate that it is a polyface mesh.

That being said, if you use code 9, IDL will run your polygons through the tessellator. I believe that this is done to make sure that the mesh is a simple triangle mesh that is appropriate for the DXF polyface mesh. The trouble is that your output from ISOSURFACE, INTERVAL_VOLUME/TETRA_SURFACE, and many other routines are already lists of triangles. Running them

through the tessellator is not necessary and in fact can mess up the mesh.
I'm not sure right now why the mesh gets corrupted. I'm looking in to it.
IDL should really avoid the tessellation if it can. (We have an open
problem report related to this)

Karl
