Subject: Re: Gridding to the Surface of a Sphere
Posted by clive.best on Mon, 20 Mar 2017 06:12:33 GMT
View Forum Message <> Reply to Message

On Monday, 15 April 2013 02:20:09 UTC+10, David Fanning wrote:
> Folks,
>
> Quite frequently you find yourself with randomly positioned data values
> that are associated with a latitude and longitude value. You often want
> to display this kind of data as a contour plot on a map projection.
> Traditionally, the Triangulate/Trigrid method is used to grid random
> data values into a 2D grid that can be contoured. And, there is
> provision in this method for gridding to the "surface of a sphere,"
> which seems like a good thing to do for latitude/longitude data.
>
> But, you would be gravely mistaken. :-)
>
> Personally, I think the Triangulate/Trigrid gridding method for creating
> a grid on the surface of a sphere is tragically flawed. (Although I
> would be happy to discover otherwise.) I have outlined in some detail my
> reasons for thinking this in the following article:
>
> http://www.idlcoyote.com/code_tips/sphericalgrid.php
>
> I also illustrate how this can be done correctly by using GridData to do
> the gridding to the sphere, rather than the Triangulate/Trigrid method.
>
> There is one strange thing about the GridData method that I don't
> understand and don't mention in the article. Maybe someone can help me
> with this. The GridData methods I illustrate (NaturalNeighbor and
> InverseDistance) require that I supply Delaunay triangles to the
> GridData program. If I create the triangles with Triangulate, all is
> well. If I create the triangles with QHull, the GridData program chokes.
> Does anyone have any insight into why that would be?
>
> You can find code and data in the article if you care to fool around
> with this.
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
> Sepore ma de ni thue. ("Perhaps thou speakest truth.")

I want to use the spherical triangulation itself rather than grid it to a regular grid. It turns out that this is a smart way to perform area averaging of global temperature data. So I really want to use the output structure Sphere=s from Triangulate. This turns out to be no easy task. IDL provides no documentation. They clearly don't want you to use it directly but pass it directly through to TRIGRID.

S is a structure with the following pattern for 1880 (much larger in later years)

```
 XYZ           DOUBLE    Array[853, 3]
   IEND          LONG      Array[853]
   IADJ          LONG     Array[5118]
```

XYZ are the cartesian coordinates on a unit sphere. However the axes bear no relation to (Lat,Lon) Latitude seems to be a linear combination of X+Y while Lon spans the z-axis.

IEND is a pointer to the last triangle for each coordinate in xyz. The triangles are defined (I think) in IADJ. Ntriangles : 5118/3 = 1706 triangles as triplet pointers into XYZ.

However when I plot the grid as triangles I get strange results. Every time I think I have solved the riddle - I get a surprise.

Has anyone got a solution to how to interpret Sphere=S ?
