Subject: Re: An easier way to draw a geodesic?
Posted by Kenneth P. Bowman on Sun, 04 Apr 2010 14:03:45 GMT
View Forum Message <> Reply to Message

In article
<018e5283-b8e4-446a-9ba1-35625dcf678c@g11g2000yqe.googlegroups.com>,
 Aram Panasenco <panasencoaram@gmail.com> wrote:

> Hello, everyone! I've been programming in IDL for a little under a
> year now, and I am currently building a program part of which involves
> graphically selecting a 4-point polygon on a sphere (each side of
> which is a geodesic).
> For that, I record the user's clicks and moves on the screen and store
> the x- and y- coordinates of the points they selected in a 2x4 array.
> The array is then processed by a function that transforms it into x-
> and y- coordinates of a spherical polygon. To do so, it first converts
> the coordinates to spherical using cv_coord. Then it uses the library
> function map_2points to find the longitude-latitude path arrays
> between the 4 point pairs. Then it combines all the longitudes and
> latitudes into one array, and cv_coord's them back into cartesian
> coordinates. The points are then used as data for an IDLgrPolyline
> object.
>
> The function works, but the resulting polygon looks extremely choppy,
> making it practically impossible to do any precision work (which is
> necessary). So my question is - how do I draw a geodesic curve without
> using three precision-degrading processes (cv_coord, map_2points, and
> cv_coord again) in a row?
>
> Thank you,
> ~Aram Panasenco
>
> P.S. I can post the function that renders the polygon online if
> necessary.

How far apart are your points? Precision should not be a problem
unless they are very close together. We use the same basic
approach all the time to draw great circles on maps (using
widget events and CONVERT_COORD).

How many points are you using to create each side of the polygon?

Ken Bowman