Subject: Re: Polygon Clipping  Algo in IDL
Posted by Mark Hadfield on Tue, 13 Dec 2005 23:02:17 GMT
View Forum Message <> Reply to Message

JD Smith wrote:

> On Tue, 13 Dec 2005 14:38:42 +1300, Mark Hadfield wrote:
>
> Hey Mark... remind me what your version did different from mine?  I recall
> spending weeks trying to vectorize this algorithm, only to give up and
> write it as a DLM.  It's fairly slow in IDL.  Lately I've been exploring
> the clipping library GPC, linking it as a DLM.

It's been a while. Didn't your POLYCLIP clip to a square? Whereas my routines clip to a single line (must be parallel to the X or Y axis for MGH_POLYCLIP and of arbitrary orientation for MGH_POLYCLIP2). Clipping to a square or rectangle therefore requires 4 applications. I settled on the line-clipping functions because they are more general and applying them 4 times is not significantly slower that applying a rectangle-clipping form once.

I spent a fair bit of time tweaking the code and achieved a modest speed-up over the code you sent me (perhaps 2x). The thing that made the difference was converting a couple of functions into in-line code.

I actually wanted this code for the following situation: I have a 2D grid of cells (usually rectangular, sometimes curvilinear) and some polygons which, for the sake of argument, we can consider to be land areas. For each cell in the grid, i want the fraction that is inside one of the polygons. So for each cell I clip every polygon and calculate the area (if any) in the clipped polygon. This approach is *not* particularly fast and implementing the polygon clipping in IDL slows it down further. I'm sure DLM-ed C code would be faster and am interested in your results.

--
Mark Hadfield          "Kei puwaha te tai nei, Hoea tahi tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)