Subject: Matching volumes and surfaces
Posted by Kenneth P. Bowman on Fri, 20 Jan 2006 02:52:34 GMT
View Forum Message <> Reply to Message

I have a 2-D data set a(x,y) and a 3-D data set b(x,y,z) that are defined on the same x and y grid.

I'm using iSurface to plot a and iVolume to overplot an isosurface of b in the same display. At the bottom of this message is an example that uses an 11 x 11 grid with data coordinates x, y, and z in the range (0, 10). The arrays a and b have been created so that a should coincide exactly with the isosurface b=0.

After running the commands below, the 2-D surface appears as a yellow plane slicing diagonally upward across the volume cube in the x-direction. Then, I choose Operations...Volume...Isosurface, change the isosurface value to 0.1, and click OK. This renders the b=0.1 isosurface in gray. (Setting the isosurface to 0.1 ensures that the isosurface is slightly above the surface a.)

The resulting image can be seen here

http://csrp.tamu.edu/hiaper/archive/render/rendering.jpg

The rather obvious problem is that the isosurface only covers the ranges x = (0, 9) and y = (0, 9).

Obviously I either don't understand the VOLUME_LOCATION and VOLUME_DIMENSIONS keywords, or I don't understand how the isosurface algorithm works, or I'm making some other mistake (the possibilities are manifold ;-)).

Can anyone steer me to the path of righteous rendering?

Thanks, Ken Bowman

```
n = 11
x = FINDGEN(n)
a = REBIN(x, n, n)

iSurface, a, x, x

xx = REBIN(x, n, n, n)
zz = REBIN(REFORM(x, 1, 1, n), n, n, n)
b  = zz - xx

iVolume, b, /OVERPLOT, $
```

VOLUME_LOCATION = [0.0, 0.0, 0.0], $
VOLUME_DIMENSIONS = [10.0, 10.0, 10.0]

Subject: Re: Matching volumes and surfaces
Posted by Karl Schultz on Mon, 23 Jan 2006 19:17:37 GMT
View Forum Message <> Reply to Message

On Thu, 19 Jan 2006 20:52:34 -0600, Kenneth P. Bowman wrote:

> I have a 2-D data set a(x,y) and a 3-D data set b(x,y,z) that are
> defined on the same x and y grid.
>
> I'm using iSurface to plot a and iVolume to overplot an isosurface of b
> in the same display. At the bottom of this message is an example that
> uses an 11 x 11 grid with data coordinates x, y, and z in the range (0,
> 10). The arrays a and b have been created so that a should coincide
> exactly with the isosurface b=0.
>
> After running the commands below, the 2-D surface appears as a yellow
> plane slicing diagonally upward across the volume cube in the
> x-direction. Then, I choose Operations...Volume...Isosurface, change
> the isosurface value to 0.1, and click OK. This renders the b=0.1
> isosurface in gray. (Setting the isosurface to 0.1 ensures that the
> isosurface is slightly above the surface a.)
>
> The resulting image can be seen here
>
> http://csrp.tamu.edu/hiaper/archive/render/rendering.jpg
>
> The rather obvious problem is that the isosurface only covers the ranges
> x = (0, 9) and y = (0, 9).
>
> Obviously I either don't understand the VOLUME_LOCATION and
> VOLUME_DIMENSIONS keywords, or I don't understand how the isosurface
> algorithm works, or I'm making some other mistake (the possibilities are
> manifold ;-)).
>

I think that the interpretation of the VOLUME_DIMENSIONS keyword is the problem.

Your surface is going to span 0 -> 10.

Part of the docs for VOLUME_DIMENSIONS say:

..., a volume with sample size of [20, 25, 20]
would render into

the region [0:19, 0:24, 0:19] in user data units.

So if you want your volume to span the same extent as the surface, you would want to use a value of 11 for VOLUME_DIMENSIONS. In this case, just don't specify it and it should do the right thing.

The code that is relevant to this discussion starts about line 330 in idlitvisisosurface__define.pro. The isosurface vertices are scaled to fit into the volume dimensions if modified by the keyword.

Karl

> Can anyone steer me to the path of righteous rendering?
>
> Thanks, Ken Bowman
>
>
> n = 11
> x = FINDGEN(n)
> a = REBIN(x, n, n)
>
> iSurface, a, x, x
>
> xx = REBIN(x, n, n, n)
> zz = REBIN(REFORM(x, 1, 1, n), n, n, n)
> b = zz - xx
>
> iVolume, b, /OVERPLOT, $
> VOLUME_LOCATION = [0.0, 0.0, 0.0], $
> VOLUME_DIMENSIONS = [10.0, 10.0, 10.0]