Subject: Re: Random spherical distributions
Posted by Kenneth P. Bowman on Sat, 01 Mar 2003 20:08:18 GMT
View Forum Message <> Reply to Message

In article <3E613888.D4C6A30D@ukc.ac.uk>, Larry Morgan <lkm8@ukc.ac.uk>
wrote:

> Hi,
> I have used the program below to create a random spherical
> distribution of particles but realised that when the distribution is
> viewed perpendicular to the z-axis a clear streak is seen running up
> down along the plot.
> Dave Fanning was contacted about this (see below) and we are both
> unsure as to why this might occur and therefore how to get around it.
> Does anyone have any suggestions of the cause of this pattern in a
> supposedly random distribution?
> cheers
> Larry
>
> Npartic=10000
>
> omega=RANDOMU(seed,Npartic,/UNIFORM,/DOUBLE) * 180.0
> rho = RANDOMU(seed1,Npartic,/UNIFORM,/DOUBLE) * 360.0
> radius = RANDOMU(seed2,Npartic,/UNIFORM,/DOUBLE)

For simplicity, think about the 2-D case on the sphere. If particles
are randomly distributed in longitude (lambda) and latitude (phi),
particles will be denser near the pole. A rectangle of given
delta-lambda and delta-phi is much smaller near the pole than near the
equator.

To fix this, you need to distribute your particles randomly in sin(phi)
rather than phi (or cos(polar angle) if you are a physicist). So
generate random numbers between -1 and 1, and then take the arcsine to
get the polar angular distribution.

The examples below use geophysical coordinates (longitude and latitude)
rather than polar angle (co-latitude) in order to use the built-in
mapping functions directly.

Your way

```
Npartic = 5000L
rho  = 360.0D0*RANDOMU(seed,Npartic,/UNIFORM,/DOUBLE)
omega= -90.0D0 + 180.0D0*RANDOMU(seed,Npartic,/UNIFORM,/DOUBLE)
MAP_SET, 90, 0, -90, /LAMBERT, /ISOTROPIC
PLOTS, rho, omega, PSYM = 3
```

Uniform way

```
   omega=!RADEG*ASIN(-1.0D0 +
2.0D0*RANDOMU(seed,Npartic,/UNIFORM,/DOUBLE))
   MAP_SET, 90, 0, -90, /LAMBERT, /ISOTROPIC
   PLOTS, rho, omega, PSYM = 3
```

The other way to do this is to generate random points uniformly distributed inside a cube, then throw away the points with radii greater than your maximum value. The problem there is getting the exact number of points that you want. ;-) (You can do it by adding random points until you reach Npartic).

Ken Bowman
