Subject: Re: How to plot great circles on a map?
Posted by Hermann Mannstein on Wed, 05 May 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Liam Gumley wrote:
>
> I seem to recall there is a way to draw great circles on a map. However
> the best I can come up with is
>
> map_set, /orth, /isot, /cont, /grid
> plots, [-45,-45], [45,-45]
>
> I really want a line between the two points that takes the shortest
> distance (in this case it would follow the longitude grid line).
>
> Any hints?
>
> ---
> Liam E. Gumley
> Space Science and Engineering Center, UW-Madison
> http://cimss.ssec.wisc.edu/~gumley

Hello Liam,
that's what I do:

```
 ;------------------------------------------------------------ --
function interp_circle,p1,p2,n     ; interpolate on a great circle
       ;  last point is p2
ll2rb,p1(1),p1(0),p2(1),p2(0),dist,azi    ; sterner's range,bearing
a_c=(indgen(n) + 1) * dist/float(n)       ; array of angles (distances)
res=fltarr(2,n)      ; array for the results
rb2ll,p1(1),p1(0),a_c,azi,lon,lat   ; from range_array,bearing to latlon
res(0,*) = lat & res(1,*) = lon
return,res
end
 ;------------------------------------------------------------ --

 ;------------------------------------------------------------ --
;+
; NAME:
;       LL2RB
; PURPOSE:
;       From latitude, longitude compute range, bearing.
; CATEGORY:
; CALLING SEQUENCE:
;       ll2rb, lng0, lat0, lng1, lat1, dist, azi
; INPUTS:
;       lng0, lat0 = long, lat of reference point (deg).    in
```

```
;       lng1, lat1 = long, lat of point of interest (deg).  in
; KEYWORD PARAMETERS:
; OUTPUTS:
;       dist = range to point point of interest (radians).  out
;       azi = azimuth to point of interest (degrees).       out
; COMMON BLOCKS:
; NOTES:
;       Notes: A unit sphere is assumed, thus dist is in radians
;         so to get actual distance multiply dist by radius.
;   Useful constants:
;   Radius of Earth (mean) = 6371.23 km = 3958.899 miles.
; MODIFICATION HISTORY:
;       R. Sterner, 13 Feb,1991
;
; Copyright (C) 1991, Johns Hopkins University/Applied Physics
Laboratory
; This software may be used, copied, or redistributed as long as it is
not
; sold and this copyright notice is reproduced on each copy made.  This
; routine is provided as is without any express or implied warranties
; whatsoever.  Other limitations apply as described in the file
disclaimer.txt.
;-
 ;------------------------------------------------------------ --

 pro ll2rb, lng1, lat1, lng2, lat2, dist, azi, help=hlp

 if (n_params(0) lt 4) or keyword_set(hlp) then begin
   print,' From latitude, longitude compute range, bearing.'
   print,' ll2rb, lng0, lat0, lng1, lat1, dist, azi'
   print,'   lng0, lat0 = long, lat of reference point (deg).    in'
   print,'   lng1, lat1 = long, lat of point of interest (deg).  in'
   print,'   dist = range to point point of interest (radians).  out'
   print,'   azi = azimuth to point of interest (degrees).       out'
   print,' Notes: A unit sphere is assumed, thus dist is in radians'
   print,'   so to get actual distance multiply dist by radius.'
   print,'   Useful constants:'
   print,'   Radius of Earth (mean) = 6371.23 km = 3958.899 miles.'
   return
 endif

 polrec3d, 1., (90.-lat2)/!radeg, lng2/!radeg, x1, y1, z1
 rot_3d, 3, x1, y1, z1, -(180.-lng1)/!radeg, x2, y2, z2
 rot_3d, 2, x2, y2, z2, -(90.-lat1)/!radeg, x3, y3, z3
 recpol3d, x3, y3, z3, r, dist, ax
 azi = (360. - ax*!radeg) mod 360.

 return
```

```
 end


 ;------------------------------------------------------------ --
;+
; NAME:
;       RB2LL
; PURPOSE:
;       From range, bearing compute latitude, longitude .
; CATEGORY:
; CALLING SEQUENCE:
;       rb2ll, lng0, lat0, dist, azi, lng1, lat1
; INPUTS:
;       lng0, lat0 = long, lat of starting point (deg).     in
;       dist = range to point of interest in RADIANS.       in
;       azi = azimuth to point of interest (degrees).       in
; KEYWORD PARAMETERS:
; OUTPUTS:
;       lng1, lat1 = long, lat of point of interest (deg).  out
; COMMON BLOCKS:
; NOTES:
;       Notes: A unit sphere is assumed, thus dist is in radians.
;         Useful constants:
;         Radius of Earth (mean) = 6371.23 km = 3958.899 miles.
;         Distance to horizon from height H above surface:
;           For small H: dist = sqrt(2*H/R) in Radians
;           For large H: dist = acos(R/(R+H)) in Radians
;         To plot horizon from lat0, lng0, H:
;           rb2ll,lng0,lat0,dist,makex(0,360,1),plng,plat
;           plots,plng-360.,plat,psym=3
; MODIFICATION HISTORY:
;       R. Sterner, 13 Feb,1991
;
; Copyright (C) 1994, Johns Hopkins University/Applied Physics
Laboratory
; This software may be used, copied, or redistributed as long as it is
not
; sold and this copyright notice is reproduced on each copy made.  This
; routine is provided as is without any express or implied warranties
; whatsoever.  Other limitations apply as described in the file
disclaimer.txt.
;-
 ;------------------------------------------------------------ --

 pro rb2ll, lng1, lat1, dist, azi, lng2, lat2, help=hlp

 if (n_params(0) lt 4) or keyword_set(hlp) then begin
   print,' From range, bearing compute latitude, longitude .'
```

```
  print,' rb2ll, lng0, lat0, dist, azi, lng1, lat1'
  print,'   lng0, lat0 = long, lat of starting point (deg).     in'
  print,'   dist = range to point of interest in RADIANS.       in'
  print,'   azi = azimuth to point of interest (degrees).       in'
  print,'   lng1, lat1 = long, lat of point of interest (deg).  out'
  print,' Notes: A unit sphere is assumed, thus dist is in radians.'
        print,'   Useful constants:'
        print,'   Radius of Earth (mean) = 6371.23 km = 3958.899
miles.'
  print,'   Distance to horizon from height H above surface:'
  print,'     For small H: dist = sqrt(2*H/R) in Radians'
  print,'     For large H: dist = acos(R/(R+H)) in Radians'
  print,'   To plot horizon from lat0, lng0, H:'
  print,'     rb2ll,lng0,lat0,dist,makex(0,360,1),plng,plat
  print,'     plots,plng-360.,plat,psym=3'
  return
 endif

 ax = (360. - azi)/!radeg
 polrec3d, 1., dist, ax, x3, y3, z3
 rot_3d, 2, x3, y3, z3, (90.-lat1)/!radeg, x2, y2, z2
 rot_3d, 3, x2, y2, z2, (180.-lng1)/!radeg, x1, y1, z1
 recpol3d, x1, y1, z1, r, az, ax
 lng2 = ax*!radeg
 lat2 = 90. - az*!radeg

 return
 end

 ;---------------------------------------------------------- -----------
```
but if you have an old IDL Version (2.0??), you will find in the
map - demo programm a routine
called cir_2p, which does the same.
--
Regards,

```
 ---------------------------------------------------------- ----
  Dr. Hermann Mannstein
                                          /|
  DLR-Institut fuer Physik der Atmosphaere        / |
  Oberpfaffenhofen                          ,---/--|---,
  D-82234 Wessling                          /   /  /   /
  Germany                                 '---|--/---'
                                        | /
  tel.:    +49-8153-28-2503                   |/  DLR
  fax.:    +49-8153-28-1841                    '
  mailto:Hermann.Mannstein@dlr.de
  http://www.op.dlr.de/ipa/          http://www.op.dlr.de/~pa64
```

---------- ----