Subject: Help plotting a 3D Carioid...
Posted by Steve[3] on Wed, 15 Feb 2006 16:46:09 GMT
View Forum Message <> Reply to Message

I'm a newbie to IDL, and am having a bit of trouble doing stuff that is pretty basic in other tools. In this case, I'm trying to generate a surface plot of a 3D cartioid function. I start by defining azimuth and elevation angle matrices, then compute the range of the cartioid for each angle pair. I then want to convert from spherical to rectangular coordinates and make a surface plot of the result.

In Matlab, I was able to do the coordinate transform on the matrices in n x n form, but it appears that IDL wants the data as column vectors. However, somewhere it appears that things are getting messed up, as the resulting plot has a "hole" in it. However, when I use xplot3D to plot the result, the hole is not there...

Additionally, the surface plot just doesn't have a nice look - is there anything a bit nicer I could do, along the lines of xplot3D, but be able to do it in a pre-specified window (I'm going to be having the result plot in a GUI)? Best would be to have the surface shaded and with the color set to correspond with the magnitude of each point.

I guess what I'm looking for is a little guidance on how to get this to look a bit nicer, and what's causing the hole when I use surface...

```
            ;ang=[0:.1:2.*pi+.1]'; (this is what I used in
Matlab...)
            ang=transpose(2*!PI*findgen(64)/63)
 n=size(ang, /N_ELEMENTS )
 ones=fltarr(n,1)+1
 theta_a=(ang##ones)
 ;ang2=[0:.05:pi+.05]'-pi/2;  (this is what Iused in Matlab)
 ang2=transpose(!PI*findgen(n)/(n-1)-!PI/2) ; (should be roughly
equivalent to above)
 theta_e=transpose(ang2##ones)
    rcart=1+cos(!pi/2-theta_e)

    theta_a_1d=reform(theta_a,1,n*n)
    theta_e_1d=reform(theta_e,1,n*n)
    rcart_1d=reform(rcart,1,n*n)
 sph=[theta_a_1d,theta_e_1d,rcart_1d]

 rectc=cv_coord(from_sphere=sph, /to_rect)
 x_1d=rectc(0,*)
 y_1d=rectc(1,*)
 z_1d=rectc(2,*)
 x=reform(x_1d,n,n)
```

```
y=reform(y_1d,n,n)
z=reform(z_1d,n,n)

        surface, z,x,y
        xplot3d, x,y,z
```

Subject: Re: Help plotting a 3D Carioid...
Posted by David Fanning on Wed, 15 Feb 2006 17:34:53 GMT
View Forum Message <> Reply to Message

Steve writes:

> Additionally, the surface plot just doesn't have a nice look - is there
> anything a bit nicer I could do, along the lines of xplot3D, but be
> able to do it in a pre-specified window (I'm going to be having the
> result plot in a GUI)? Best would be to have the surface shaded and
> with the color set to correspond with the magnitude of each point.
>
> I guess what I'm looking for is a little guidance on how to get this to
> look a bit nicer, and what's causing the hole when I use surface...

Hard to say what is happening to SURFACE. I note that
rotating the surface slightly gets around the problem:

```
  surface, z, x, y, az=45
```

This is probably an artifact from the fact that it is
really a 2.5D representation of a surface. But, it looks
like a bug, for sure.

XPlot3D uses an object graphics surface (IDgrSurface), rather
than a direct graphics surface. This is a true 3D surface,
so I am not surprised this works correctly. You can
put an object graphics surface into an object graphics
window, but the whole process is quite a bit more
trouble than it is with direct graphics. :-)

If this is something you want to pursue, I'd recommend
getting ahold of Ronn Kling's book, Power Graphics in IDL.
The chances of learning to use object graphics from
the IDL documentation is slim and none, I believe, unless
you have some graphics experience from somewhere else.

Cheers,

David
--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Help plotting a 3D Carioid...
Posted by Steve[3] on Wed, 15 Feb 2006 22:36:41 GMT
View Forum Message <> Reply to Message

Thanks for your prompt and detailed reply... I just checked out your extensive website, and your fsc_surface routine is very close to what I'm looking for... However, I'm not sure if I can use it with repeated calls.

Do you (or anyone else on here) know if it is possible (within the GUI) to surf something in a separate object window with fsc_surface (or XPlot3D), then when I push another button on the GUI, have the first object window close, and open a new one for the second plot (or just overwrite the old with the new)?

Also, is it possible to simply have this object window display in the GUI by replacing the base with a specified base?

Thanks...

-Steve

P.S. The following is along the lines of how I am attempting the plot...

```
pro ant_test_event, event
   widget_name=widget_info(event.id, /uname)
   ;print, widget_name

   case widget_name of


  'pushit': begin
   ;print, 'in fslider1_roll'
    widget_control, event.top, get_uvalue=global_data

   ;ang=[0:.1:2.*pi+.1]';
 ang=transpose(2*!PI*findgen(14)/13)
 n=size(ang, /N_ELEMENTS )
 ones=fltarr(n,1)+1
 theta_a=(ang##ones)
 ;ang2=[0:.05:pi+.05]'-pi/2;
```

```
  ang2=transpose(!PI*findgen(n)/(n-1)-!PI/2)
  theta_e=transpose(ang2##ones)
     r_o=fltarr(n,n)+1
     rcart=1+cos(!pi/2-theta_e)
     ;rcart=r_o

     theta_a_1d=reform(theta_a,1,n*n)
     theta_e_1d=reform(theta_e,1,n*n)
     rcart_1d=reform(rcart,1,n*n)
  sph=[theta_a_1d,theta_e_1d,rcart_1d]

  rectc=cv_coord(from_sphere=sph, /to_rect)
  x_1d=rectc(0,*)
  y_1d=rectc(1,*)
  z_1d=rectc(2,*)
  x=reform(x_1d,n,n)
  y=reform(y_1d,n,n)
  z=reform(z_1d,n,n)

        fid=global_data.window1_id
        wset, fid
        surface, z,x,y;, AX=45, AZ=45
        ;shade_surf, z;,x,y;AX=-30, AZ=-45
        ;xplot_sl, x,y,z
        ;fsc_surface, z,x,y
   end


          'Quit': begin
            print,'Got into Quit Case'
            widget_control, event.top, /destroy
          end
    else:
    endcase
end

 ;------------------------------------------------------------ ------------------------------------

pro ant_test
 fund_base=widget_base(column=2, title="Plotting test GUI")

    N=256
    draw1=widget_draw(fund_base, xsize=N, ysize=N, uname="window1")
    pushit=widget_button(fund_base, value="Push to Plot",
uname="pushit")
 quit_button=widget_button(fund_base, value="Quit", uname="Quit")
    widget_control, fund_base, /realize
    widget_control, draw1, get_value=window1
```

```
   global_data={draw1_id: draw1, window1_id: window1 }
   widget_control, fund_base, set_uvalue=global_data
   xmanager, 'ant_test', fund_base
   event_handler='ant_test_event'
end
```

Subject: Re: Help plotting a 3D Carioid...
Posted by David Fanning on Wed, 15 Feb 2006 23:29:53 GMT
View Forum Message <> Reply to Message

Steve writes:

> Thanks for your prompt and detailed reply... I just checked out your
> extensive website, and your fsc_surface routine is very close to what
> I'm looking for... However, I'm not sure if I can use it with repeated
> calls.

It was not designed for that purpose, no.

> Do you (or anyone else on here) know if it is possible (within the GUI)
> to surf something in a separate object window with fsc_surface (or
> XPlot3D), then when I push another button on the GUI, have the first
> object window close, and open a new one for the second plot (or just
> overwrite the old with the new)?
>
> Also, is it possible to simply have this object window display in the
> GUI by replacing the base with a specified base?

Everything is possible in IDL. Well, excluding truly device-independent programs, of course. But everything within the limited imagination of astronomers, surely. :-)

The problem is, how to do it. :-(

What you like about IDL (I can tell) is how easy it is to use. You type a command, you see a result in a graphics window. Brilliant! But that only works in what we call "direct graphics." There is another whole (and completely different and incompatible) graphics system in IDL that is MUCH more powerful and MUCH more difficult to use. It is called the "object graphics" system.

The FSC_Surface program appears to do what the SURFACE command does (plus a bit more), but it is MUCH more complicated than that. You write a least a page of code in the object graphics system to replicate what is already done for you in the SURFACE command. And then

you have to display object graphics output in an object graphics window. You can't mix and match direct graphics commands and output with object graphics commands and output.

You can't just willy-nilly throw FSC_Surface output in any window, because the *interaction* with the surface depends on the window. (If you don't like the rotations, etc., you could, of course, just throw a static rendering of your surface in the window. This is about 3/4 of a page of code, but easily done. You just have to build the surface, the axes, rotate everything into your view, establish the view, stuff everything into a model, etc., etc. It really isn't as bad as it sounds. See Simple_Surface as a stripped down example of FSC_Surface.)

And, of course, what you do in a widget program is completely divorced from the graphics system you are using. Although you have to know how to create draw widgets that represent windows in one system or the other, depending on which you are using. But that is trivial.

So, where does that leave you? At the moment it leaves you writing direct graphics code (WSET, SURFACE, etc.) and wishing you knew how to write object graphics code (IDLgrWindow, IDLgrSurface, IDLgrAxis, etc, etc.). I'd get Ronn's book. Otherwise, you will be at this a LONG time. Maybe longer than you have the stomach for it. Especially as you are coming from a Matlab background. You can't be thinking pleasant thoughts about IDL at the moment. :-)

Cheers,

David

P.S. There is another possibility. You could try iTools. Almost none of us here can figure out how to program the damn things, but there is always the small change they work the way you want them to work. Try iSurface. Let us know how it goes. :-)

P.S.S. If it is any consolation to you, I feel for you, I really do.

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
