Subject: Re: Function Graphics Questions
Posted by Paul Van Delst[1] on Tue, 14 Jan 2014 21:23:00 GMT
View Forum Message <> Reply to Message

On 01/14/14 12:56, David Fanning wrote:
> Folks,
>
> I have a New Year's resolution to learn more about function graphics
> this year. I thought I would start with this multiple axis plot Chris
> and Matt have been talking about this week. My idea was to produce
> mirror of the Coyote Gallery plots, as much as I can.
>
> I guess I've spent about three hours on this now and finally have this
> one program in shape to be able to compare Coyote Graphics output with
> the equivalent function graphics output. In doing so, I've run into some
> questions. Perhaps someone knows the answers.
>
> Question 1: I have no particular objection to the PostScript output
> produced by function graphics commands, but is it true there is no
> program control over things like the thickness of the PostScript lines?

?

You mean you don't see any difference when you use the THICK keyword?

I see a difference in the line thickness between ps output when I do:

```
p = Plot(/test, LineStyle=2)
p.save, 'test.ps'
```

and

```
p = Plot(/test, LineStyle=2,thick=2)
p.save, 'test.ps'
```

> Question 2: As far as I can tell, saving the contents of a function
> graphics window as a PostScript file *always* creates encapsulated
> PostScript files. Since encapsulated Postscript files (AFAIK) always
> have to be in portrait mode, what is the purpose of the LANDSCAPE
> keyword to the window save command?
>
> In other words, this command:
>
>   window.save, 'test.ps', /Landscape
>
> Produces exactly the same output, as far as I can tell, as this command:

>
>   window.save, 'test.ps'

Dunno. I always create png output or eps for including in documents so I
don't use /landscape.

> Question 3: I haven't upgraded to IDL 8.3. Can someone tell me if the
> bug in IDL 8.2.3 that prevents any line style except solid in PostScript
> output is fixed. In other words, do these commands produce a Postscript
> plot with a dashed line:
>
>   p = Plot(/test, LineStyle=2)
>   p.save, 'test.ps'

IDL v8.3 produces a dashed line in the ps output.

> Question 4: Am I missing something obvious here. I mostly produce JPEG,
> PNG, and TIFF output either for my web page of for e-mailing
> intermediate results to colleagues. I like them to be reasonably small.
> For my web page, for example, I like them to be no more than 600 pixels
> wide. My usual way of creating such raster output is to run my code like
> this:
>
>   cgPS_Open, 'test.png'
>   cgPlot, cgDemoData(1)
>   cgPS_Close, Width=600
>
> The equivalent in function graphics is something like this:
>
>   p = Plot(cgDemoData(1))
>   p.save, 'test.png', width=600
>
> But, this kind of output is very low resolution compared to what I've
> come to expect.

I don't know if it's the same thing (you're windows, right?), but I was
experiencing a problem with v8.2.2/3 on a RHEL6 system where the png
output of a plot had very blocky/pixel-y lines. Exelis help replicated
the problem on their CentOS systems even with v8.3. The following
workaround provided by Exelis made the lines smooth again:

```
$ export IDL_DISABLE_STROKED_LINES=1
```

The issue has been reported as IDL-69024.

cheers,

paulv

>
> I find the only way I can get high quality PNG files is to produce them
> at full resolution, then resize them in the software I use for dealing
> with raster images (Photoshop, Hypersnap, etc.). Since I have
> ImageMagick hanging around, I find I can get what I want in IDL by doing
> something like this:
>
>   p = Plot(cgDemoData(1))
>   p.save, 'test.png'
>   Spawn, 'convert test.png -resize 600 test_resized.png'
>
> Is there a better way to do this?
>
> Cheers,
>
> David
>