Subject: Re: Plotting functions in IDL
Posted by M.Reuss on Wed, 17 Jan 1996 08:00:00 GMT
View Forum Message <> Reply to Message

For plotting, IDL needs a vector of points to plot.
Therefore you evaluate your function at enough points and
then plot this vector.

Suppose the function MYFUNCTION can be evaluated at one point only
in each call and you want to plot it in the range [20,140.], one point
at each integer number being sufficient for you. Then the following should do
the job:

```
xarr = FINDGEN(121) + 20.
yarr = FLTARR(121)
FOR i = 0,N_ELEMENTS(xarr)-1 DO yarr(i) = MYFUNCTION(xarr(i))

PLOT,xarr,yarr
```

This will look even more simple if the function code allows to handle
a whole array in one call.

I am afraid I haven't understood what you mean with 'no equations'. You can
code in IDL everything which you can code in FORTRAN, all numerical math
works with IDL (there can be a performance problem, though).
If you mean computer algebra, that's something very different...

Matthias Reuss

Subject: Re: Plotting functions in IDL
Posted by Liam Gumley on Wed, 17 Jan 1996 08:00:00 GMT
View Forum Message <> Reply to Message

kunstman@pu.informatik.th-darmstadt.de (Thomas Kunstmann) wrote:
> What is the best way of plotting a function in IDL? As far as I know,
> it only handles vectors of discrete data and no equations.

How about the following to plot sin(x) over 0 to 2*pi radians:

```
xmin = 0.0
xmax = 2.0 * !pi
n = 1000
x = ( findgen( n ) / float( n - 1 ) ) * ( xmax - xmin ) + xmin
plot, x, sin(x)
```

Use as many points (n) as you like to make a smooth curve.

Cheers,
Liam.