
Subject: Re: "TRUE" map projections

Posted by [zawodny](#) on Fri, 27 Jan 1995 17:32:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article <3gb5au\$03i@post.gsfc.nasa.gov> Serdar Manizade <manizade@osb1.wff.nasa.gov> writes:

> I repost my note since subject was missing. Apologies.

> Serdar Manizade <manizade@osb1.wff.nasa.gov> wrote:

>>

>> [zawodny@arbd0.larc.nasa.gov](#) (Joseph M Zawodny) wrote:

>>> possible I use "circular" map projections and the SET_ISOXY routine. But,

>>

>> Where is the SET_ISOXY routine?

>>

>> When I produce a map, I have to pull out my ruler and measure the

>> latitude and longitude scales to see if I got the scaling right. I

>> make adjustments whenever I change my output map aspect ratio or the

>> coordinates of the map. Especially important at high latitudes.

>

This is my version of the routine set_isoxy. It was part of a system of routines written by the original author. I hacked it a bit to work by itself without the need for the other routines. It works for what I use it for but it may no longer function as the original author intended. I make no claims of functionality for either myself or the original author. Look it over, try it out and hack it further to suit your needs.

PRO SET_ISOXY, XMN0, XMX0, YMN0, YMX0

;+

;-----

; Name: SET_ISOXY.PRO

; Purpose: Set XY isotropically (equal x and y scale).

; Category: Graphics.

; Calling sequence: SET_ISOXY, XMN, XMX, YMN, YMX

; Inputs:

; XMN = minimum data x. in.

; XMX = maximum data x. in.

; YMN = minimum data y. in.

; YMX = maximum data y. in.

; Optional input parameters:

; Outputs:

; Optional output parameters:

; Common blocks: ISOXY_COMMON = what parts of data window to lock in place.

; Side effects:

; Restrictions:

; Routines used:

; Procedure:

```

; Modification history:
; 3 Sep, 1986. R. Sterner. Johns Hopkins U APL
; Jun 14, 1990. J.M. Zawodny NASA LaRC modified for standalone to V-2.0
;-----
;-
COMMON ISOXY_COMMON, X_LOCK, Y_LOCK

IF N_PARAMS(0) EQ 0 THEN BEGIN
  PRINT, 'SET_ISOXY, XMN, XMX, YMN, YMX
  PRINT, ' XMN, XMX = desired min and max X.
  PRINT, ' YMN, YMX = desired min and max Y.
  PRINT, ' Either XMN, XMX or YMN, YMX will be adjusted to force equal scaling in
  PRINT, ' X and Y in the set screen window. At least the specified range will
  PRINT, ' be covered in both X and Y, but a greater range will be covered in one.
  RETURN
ENDIF

IF N_ELEMENTS(X_LOCK) EQ 0 THEN X_LOCK = 'XMD' ; defaults.
IF N_ELEMENTS(Y_LOCK) EQ 0 THEN Y_LOCK = 'YMD'
XMN = XMN0 ; copy so changing values
XMX = XMX0 ; won't change them in caller.
YMN = YMN0
YMX = YMX0

; if !p.position not set default to largest area possible
m=where(!p.position ne 0.,count)
if(count eq 0) then !p.position=[0.,0.,1.,1.]

IDX = (!p.position(2)-!p.position(0))*!d.x_vsize ; plot position
IDY = (!p.position(3)-!p.position(1))*!d.y_vsize ; side lengths
DX = FLOAT(XMX - XMN) ; data window side lengths.
DY = FLOAT(YMX - YMN)
YMD = .5*(YMN + YMX) ; data window midpoint.
XMD = .5*(XMN + XMX)
IF (DX/DY) GE (IDX/IDY) THEN BEGIN ; adjust Y range.
  DY2 = .5*DX*IDY/IDX ; half new Y range.
  DX2 = .5*DX ; half old X range.
ENDIF ELSE BEGIN ; adjust X range.
  DX2 = .5*DY*IDX/IDY ; half new X range.
  DY2 = .5*DY ; half old Y range.
ENDELSE
;----- lock window -----
CASE X_LOCK OF ; lock x.
'XMN': XMX = XMN + DX2 + DX2 ; lock x min.
'XMD': BEGIN ; lock x mid.
  XMN = XMD - DX2
  XMX = XMD + DX2
END

```

```
'XMX': XMN = XMX - DX2 - DX2 ; lock x max.  
ENDCASE  
CASE Y_LOCK OF ; lock y.  
'YMN': YMX = YMN + DY2 + DY2 ; lock y min.  
'YMD': BEGIN ; lock y mid.  
    YMN = YMD - DY2  
    YMX = YMD + DY2  
END  
'YMX': YMN = YMX - DY2 - DY2 ; lock y max.  
ENDCASE
```

```
!x.range=[XMN, XMX]  
!y.range=[YMN, YMX]
```

```
RETURN  
END
```

--

```
Joseph M. Zawodny (KO4LW)          NASA Langley Research Center  
Internet: j.m.zawodny@larc.nasa.gov  MS-475, Hampton VA, 23681-0001  
TCP/IP: ko4lw@ko4lw.ampr.org Packet: ko4lw@n4hog.va.usa.na
```
