Subject: Re: Scaling of axes 1:1?
Posted by thompson on Tue, 10 Oct 1995 07:00:00 GMT
View Forum Message <> Reply to Message

heide@immr.tu-clausthal.de (Gerhard Heide) writes:

> Hi,

> who can I scaling the axes in a contuor plot 1:1?
> The tick interval is perhaps 1m of the x and y axes, but the printed plot
> isn't it equal intervals.

> I can scale the contuor plot with postscript options XSIZE and YSIZE, but
> the plot is depented from the axes label and title!

This routine may do what you want to do.

Bill Thompson

```
 ============================================================
==================
 PRO SETSCALE,P1,P2,P3,P4,NOBORDER=NOBORDER,NOADJUST=NOADJUST
;+
; Project     : SOHO - CDS
;
; Name        :
; SETSCALE
; Purpose     :
; Sets plot scale so it is the same in X and Y directions.
; Explanation :
; The data limits in the X and Y directions (plus 5%) are calculated and
; compared against the the physical size of the plotting area in device
; coordinates.  Whichever scale is larger is then used for both axes, and
; the plot limits are set to center the data in both directions.  The
; parameters !X.STYLE and !Y.STYLE are then set to 1 for exact spacing.
; Use         :
; SETSCALE    - Resets to previous state.
; SETSCALE, ARRAY    - Calculates scale for CONTOUR.
; SETSCALE, XARRAY, YARRAY  - Calculates scale from arrays.
; SETSCALE, XMIN, XMAX, YMIN, YMAX - Calculates scale from limits.
; Inputs      :
; None required.  Calling SETSCALE without any parameters resets to the
; default behavior.
; Opt. Inputs :
; ARRAY   - Two dimensional array to be used in a simple
;       contour plot.  The minima are set to zero,
;       and the maxima are set to one less than the
;       dimensions of the array.
```

```
; XARRAY, YARRAY  - Arrays from which the minimum and maximum
;      values are calculated.
; XMIN, XMAX, YMIN, YMAX - The limits in the X and Y directions from
;      which the scale is calculated.  The actual
;      X and Y ranges must include these values.
; Outputs     :
; None.
; Opt. Outputs:
; None.
; Keywords    :
; NOBORDER = If set, then the 5% border is not applied.
; NOADJUST = If set, then the edges of the plot (!P.POSITION) are
;     not modified.
; Calls       :
; GET_VIEWPORT
; Common      :
; SETSCALE = Keeps track of the system variables changed by this routine.
; Restrictions:
; Unpredictable results may occur if SETSCALE is in effect when WINDOW,
; WSET or SET_PLOT are called.  It is recommended that SETSCALE be called
; without parameters to reset to the ordinary behavior before the
; graphics device or window is changed.
;
; In general, the SERTS graphics devices routines use the special system
; variables !BCOLOR and !ASPECT.  These system variables are defined in
; the procedure DEVICELIB.  It is suggested that the command DEVICELIB be
; placed in the user's IDL_STARTUP file.
;
; Side effects:
; The system variables !X.STYLE, !Y.STYLE, !X.S, !Y.S, !X.RANGE (!XMIN
; and !XMAX) and !Y.RANGE (!YMIN and !YMAX) are modified.
;
; Unless NOADJUST is set, the edges of the plot (!P.POSITION) are
; adjusted to fit the data.  Then, when SETSCALE is called without any
; parameters, these parameters are returned to their original settings.
;
; System variables may be changed even if the routine exits with an error
; message.
;
; If SETSCALE is called without any parameters, then the modified system
; variables are restored to their original values.  Additional graphics
; functions such as OPLOT will still be possible.
;
; Category    :
; Utilities, Devices.
; Prev. Hist. :
; William Thompson, Feb. 1991.
; William Thompson, Oct. 1991, added !ASPECT system variable.
```

```
; William Thompson, May  1992, added common block and changing viewport.
; William Thompson, Nov. 1992, changed structure of common block, and
;  removed support for changing viewport.
; William Thompson, Nov. 1992, changed to use GET_VIEWPORT instead of
;  INIT_SC1_SC4, and to restore original !P.POSITION when called
;  with no parameters.
; William Thompson, December 1992, changed common block to keep better
;  track of the state of the system variables.
; Written     :
; William Thompson, GSFC, February 1991.
; Modified    :
; Version 1, William Thompson, 27 April 1993.
;  Incorporated into CDS library.
; Version     :
; Version 1, 27 April 1993.
;-
;
 ON_ERROR,2
 COMMON  SETSCALE,POS_SET,SCL_SET,XSTYLE,YSTYLE,XRANGE,YRANGE,POSITIO N
;
;  Check to see if the common block variables have been initialized.
;
 IF N_ELEMENTS(POS_SET) EQ 0 THEN BEGIN
  POS_SET = 0
  SCL_SET = 0
  XSTYLE  = !X.STYLE
  YSTYLE  = !Y.STYLE
  XRANGE = !X.RANGE
  YRANGE = !Y.RANGE
  POSITION = !P.POSITION
 ENDIF
;
;  Check to see if the screen coordinates have been stored.  If they have
;  already been set, then reset them to their original settings so that the
;  routine starts with a clean plate.
;
 IF POS_SET EQ 0 THEN BEGIN
  IF SCL_SET EQ 0 THEN BEGIN
   XSTYLE = !X.STYLE
   YSTYLE = !Y.STYLE
   XRANGE = !X.RANGE
   YRANGE = !Y.RANGE
  ENDIF
  POSITION = !P.POSITION
 END ELSE IF NOT KEYWORD_SET(NOADJUST) THEN BEGIN
  !P.POSITION = POSITION
  GET_VIEWPORT,SC1,SC2,SC3,SC4
  !X.CRANGE = ([SC1,SC2]/!D.X_SIZE - !X.S(0)) / !X.S(1)
```

```
  !Y.CRANGE = ([SC3,SC4]/!D.Y_SIZE - !Y.S(0)) / !Y.S(1)
  POS_SET = 0
 ENDIF
;
;  If no parameters were passed, then reset the range parameters and return.
;
 IF N_PARAMS() EQ 0 THEN BEGIN
  IF SCL_SET NE 0 THEN BEGIN
   !X.STYLE = XSTYLE
   !Y.STYLE = YSTYLE
   !X.RANGE = XRANGE
   !Y.RANGE = YRANGE
   SCL_SET = 0
  ENDIF
  RETURN
;
;  If only one parameter was passed, then set the scale as it would be used for
;  CONTOUR.
;
 END ELSE IF N_PARAMS() EQ 1 THEN BEGIN
  SZ = SIZE(P1)
  IF SZ(0) NE 2 THEN BEGIN
   PRINT,'*** ARRAY must be two dimensional, ' + $
    'routine SETSCALE.'
   RETURN
  ENDIF
  XMIN = 0  &  XMAX = SZ(1) - 1
  YMIN = 0  &  YMAX = SZ(2) - 1
;
;  If two parameters are passed, then set the scale for PLOT,X,Y
;
 END ELSE IF N_PARAMS() EQ 2 THEN BEGIN
  XMIN = MIN(P1, MAX=XMAX)
  YMIN = MIN(P2, MAX=YMAX)
;
;  If four parameters are passed, then the ranges have been passed directly.
;
 END ELSE IF N_PARAMS() EQ 4 THEN BEGIN
  XMIN = P1  &  XMAX = P2
  YMIN = P3  &  YMAX = P4
;
;  Otherwise, print an error message.
;
 END ELSE BEGIN
  PRINT,'*** SETSCALE must be called with 0-4 parameters:'
  PRINT,'        SETSCALE'
  PRINT,'        SETSCALE, ARRAY'
  PRINT,'        SETSCALE, XARRAY, YARRAY'
```

```
  PRINT,'        SETSCALE, XMIN, XMAX, YMIN, YMAX'
  RETURN
 ENDELSE
;
;  Check to see if the screen coordinates have been initialized.
;
 GET_VIEWPORT,SC1,SC2,SC3,SC4
;
;  Calculate the plot scale.
;
 XSCALE = ABS(XMAX - XMIN) / (SC2 - SC1) * !ASPECT
 YSCALE = ABS(YMAX - YMIN) / (SC4 - SC3)
 IF NOT KEYWORD_SET(NOBORDER) THEN BEGIN
  XSCALE = 1.05 * XSCALE
  YSCALE = 1.05 * YSCALE
 ENDIF
 SCALE = XSCALE > YSCALE
 IF SCALE LE 0 THEN BEGIN
  PRINT,'*** Unable to calculate the plot scale, routine SETSCALE.'
  RETURN
 ENDIF
;
;  Calculate the new screen coordinates.
;
 IF NOT KEYWORD_SET(NOADJUST) THEN BEGIN
  XAVG = (SC1+SC2) / 2.
  YAVG = (SC3+SC4) / 2.
  XDELTA = ABS(XMAX - XMIN) / SCALE * !ASPECT
  YDELTA = ABS(YMAX - YMIN) / SCALE
  IF NOT KEYWORD_SET(NOBORDER) THEN BEGIN
   XDELTA = 1.05 * XDELTA
   YDELTA = 1.05 * YDELTA
  ENDIF
  SC1 = XAVG - XDELTA / 2.  &  SC2 = XAVG + XDELTA / 2.
  SC3 = YAVG - YDELTA / 2.  &  SC4 = YAVG + YDELTA / 2.
  !SC1 = SC1  &  !SC2 = SC2
  !SC3 = SC3  &  !SC4 = SC4
  POS_SET = 1
 ENDIF
;
;  Calculate the edges of the plot.
;
 XAVG = (XMIN+XMAX) / 2.
 YAVG = (YMIN+YMAX) / 2.
 XDELTA = SCALE * (SC2-SC1) / (2*!ASPECT)
 YDELTA = SCALE * (SC4-SC3) / 2.
 !X.RANGE = [XAVG - XDELTA, XAVG + XDELTA]
 !Y.RANGE = [YAVG - YDELTA, YAVG + YDELTA]
```

```
 SCL_SET = 1
;
;  Check to see if the data should be plotted in reverse order.
;
 IF XMIN GT XMAX THEN !X.RANGE = REVERSE(!X.RANGE)
 IF YMIN GT YMAX THEN !Y.RANGE = REVERSE(!Y.RANGE)
;
;  Set the style parameters to force the data to be plotted with the exact
;  edges calculated.
;
 !X.STYLE = !X.STYLE OR 1
 !Y.STYLE = !Y.STYLE OR 1
;
;  Calculate the scaling parameters !X.S and !Y.S
;
 !X.S(1) = (SC2 - SC1) / (!X.RANGE(1) - !X.RANGE(0)) / !D.X_SIZE
 !Y.S(1) = (SC4 - SC3) / (!Y.RANGE(1) - !Y.RANGE(0)) / !D.Y_SIZE
 !X.S(0) = !X.WINDOW(0) - !X.RANGE(0)*!X.S(1)
 !Y.S(0) = !Y.WINDOW(0) - !Y.RANGE(0)*!Y.S(1)
;
 RETURN
 END
 ================================================================
==================
 PRO GET_VIEWPORT, SC1, SC2, SC3, SC4
;+
; Project     : SOHO - CDS
;
; Name        :
; GET_VIEWPORT
; Purpose     :
; Gets current viewport values, in device coordinates.
; Explanation :
; Gets the current values of the viewport, in the form of the
; old-fashioned variables !SC1, !SC2, !SC3, and !SC4.  This supports
; those routines that were originally developed for IDL version 1.
;
; The routine calculates the system variables by generating a dummy plot
; without actually drawing to the screen.
;
; Use         :
; GET_VIEWPORT, SC1, SC2, SC3, SC4
; Inputs      :
; None.
; Opt. Inputs :
; None.
; Outputs     :
; SC1, SC2, SC3, SC4 are the device coordinates of the viewport.
```

```
; Opt. Outputs:
; None.
; Keywords    :
; None.
; Calls       :
; None.
; Common      :
; None.
; Restrictions:
; In general, the SERTS graphics devices routines use the special system
; variables !BCOLOR and !ASPECT.  These system variables are defined in
; the procedure DEVICELIB.  It is suggested that the command DEVICELIB be
; placed in the user's IDL_STARTUP file.
;
; Side effects:
; None.
; Category    :
; Utilities, Devices.
; Prev. Hist. :
; William Thompson, November 1992.
; William Thompson, November 1992, modified to get parameters by
;  generating a dummy plot rather than calculating directly, so as
;  to be compatible with !P.MULTI.
; William Thompson, December 1992, corrected bug where certain system
;  variables were being changed by this routine.
; Written     :
; William Thompson, GSFC, November 1992.
; Modified    :
; Version 1, William Thompson, GSFC, 27 April 1993.
;  Incorporated into CDS library.
; Version 2, William Thompson, GSFC, 12 January 1994.
;  Modified to avoid problems that may arise when !X.STYLE or
;  !Y.STYLE is not zero.
; Version     :
; Version 2, 12 January 1994.
;-
;
 ON_ERROR, 2
;
;  Check the number of parameters.
;
 IF N_PARAMS() NE 4 THEN MESSAGE, $
  'Syntax:  GET_VIEWPORT, SC1, SC2, SC3, SC4'
;
;  Save the current settings of the system variables.
;
 P = !P  &  X = !X  &  Y = !Y  &  Z = !Z
;
```

```
;  Do a dummy plot.
;
  PLOT,[0,0],[1,1],/NODATA,XSTYLE=4,YSTYLE=4,TITLE='',/NOERASE
;
;  Get the values of !SC1, etc.
;
 SC1 = !SC1  &  SC2 = !SC2  &  SC3 = !SC3  &  SC4 = !SC4
;
;  Restore the system variables.
;
 !P = P  &  !X = X  &  !Y = Y  &  !Z = Z
;
 RETURN
 END
```