

## 16.3 Plotting Using T<sub>E</sub>X

### A. Purpose

These subprograms produce plots for documents prepared using the T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X languages.

### B. Usage

To produce a plot using T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X, one constructs a program that uses SPLOT. SPLOT writes a file of T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X commands. The file is then incorporated into a T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X document, which is then processed by a T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X processor. Detailed instructions are given in Section B.5 below.

Sections B.1 through B.6 below describe:

B.1	Program Prototype, Single Precision . . . . .	1
B.2	Argument Definitions . . . . .	1
B.3	Numerical Option Descriptions . . . . .	1
B.4	Textual Option Descriptions . . . . .	7
B.5	Incorporating the Output Into a T <sub>E</sub> X or L <sup>A</sup> T <sub>E</sub> X Document . . . . .	9
B.6	Modifications for Double Precision . . . . .	9

#### B.1 Program Prototype, Single Precision

**INTEGER** NX

**REAL** XSIZE, YSIZE, X( $\geq$  NX), Y( $\geq$  NX),  
OPT(see below)

**CHARACTER** COPT(see below)

Assign values to NX, XSIZE, YSIZE, X, Y, OPT and COPT, and use the following subroutine reference.

**CALL SPLOT (XSIZE, YSIZE, X, NX, Y,  
OPT, COPT)**

At this point, T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X commands to produce the plot are in the file `splot.tex` (unless another file name is selected by options).

#### B.2 Argument Definitions

**XSIZE** [in] Physical horizontal size of the drawing area for the plot. Default units are inches.

**YSIZE** [in] Physical vertical size of the drawing area for the plot. Default units are inches. The nominal drawing area for the plot is a region of size XSIZE  $\times$  YSIZE. The origin of the physical coordinate system is at the lower left corner, and all border lines are on the boundary of this region. Physical coordinates outside this area may be specified. The actual area of the image is increased by outward-pointing “tick” marks (see option 5 in Section B.3), “tick” labels (see

options 4 and 5 in Section B.3) or annotations placed outside the drawing area (see option 14 in Section B.3).

Many of the options refer to user coordinates that are specified in the same units as the points that are plotted. Different user coordinates may be used in a plot, see option 7 in Section B.3.

**X** [in] Array of NX abscissas.

**NX** [in] Number of abscissas in X and ordinates in Y.

**Y** [in] Array of NX ordinates. See also option 2 in Section B.3.

**OPT** [inout] OPT(1) is status output. Nonzero values indicate a problem for which an error message is printed. Possibilities are listed in the comments of subroutine SPLOTE in the file `splot.f` or `splot.for`. Unless one has taken special action by calling MESS, see Chapter 19.3, all except for 6 warning messages stop execution. Values from 1–6 are for warnings, 10–19 are for problems in COPT, 20–32 are for problems in OPT, values  $\geq 40$  are caused by problems that probably are due to bugs in this plotting software that should be fixed.

Beginning in OPT(2), the user provides option specifications as described in Section B.3.

**COPT** [in] Character options, and data used by options in OPT, as described in Section B.4.

For simplest usage, set OPT(2) = 0 and COPT = ‘Q’.

#### B.3 Numerical Option Descriptions

Options are specified by providing a sequence of codes starting in OPT(2), and by sequences of characters in COPT().

For options specified in OPT(), each code is a whole number, followed by zero or more “arguments.”

Arguments are numbers, which depending on the option, may be whole numbers. They may be data or indices into the array COPT() indicating the location of a character datum. Where we write (Argument X=n) below, n is the default value for X if the option is not selected.

“Whole numbers” in OPT() are determined by using the Fortran nint() intrinsic function, that is, the nearest whole number to OPT(j) is used if OPT(j) is expected to be a whole number.

If an option code is negative, say OPT(j) < 0, then OPT(j) is not processed, but the number of “arguments” used by option |OPT(j)| are skipped.

## Summary of Options

Option or Field	Brief Description	Scratch File	Remarks, Other affected options
0	End of options.		
A1/10 <sup>0</sup>	Finish or not, set units		Subsequent data
A1/10 <sup>1</sup>	Interpolation	Y	Subsequent data
A1/10 <sup>2</sup>	log <sub>10</sub> on X, (polar – not implemented)	Y	Subsequent data, 14–16, 19–21
A1/10 <sup>3</sup>	log <sub>10</sub> on Y or polar info.	Y	Subsequent data, 14–16, 19–21
A1/10 <sup>4</sup>	Lines at X = 0, Y = 0		
A1/10 <sup>5</sup>	TeX or L <sup>A</sup> TeX		Use final value
2	For multiple curves	Y	Required by some cases in option 10
3	Solid, dashed, dotted curves, width, etc.	Y	Subsequent data, 19, 22
4	Border characteristics		Use final value for each border or axis
5, 6	Length and spacing of border tick marks		Use final value for each border or axis
7	Change current data set	Y	data, 8, 9, 4, 11, 14, 15, and 19 – 24
8, 9	Adjust user coordinates, clipping		Digits 10 <sup>3:4</sup> of S4
10	Plot symbols, error bars, vectors	Y	Subsequent data
11	Plot a single symbol, etc.	Y	Nothing
12	Draw arrow heads on curves	Y	Subsequent data, 19, 22
13	Set various line widths		
A13–D13	Line widths for border, tick, etc.		Use final value
E13	Line widths for rectangles and ellipses	Y	20, 21, 23, and 24
14	Text annotation	Y	Nothing
15	Put a number on the plot	Y	Nothing
16	Extra tick / annotation	Y	Nothing
17	Bad data value		Use final value
18	Not used.		
19	Draw a line	Y	Nothing
20	Draw a rectangle	Y	Nothing
21	Draw an ellipse	Y	Nothing
22–24	As for 19–21, but in physical coordinates	Y	Nothing
25	Type of fill for closed regions	Y	Subsequent data, 20, 21, 23, 24
26	Not used.		
27	Debug output		Final value per call, then reset

The third and fourth columns of the table are described in Section D.1.

**1** (Argument N1 = 0) N1 is a whole number in which the decimal digits specify options. Digits of N1 mean:

### Number Description

**0** No more options. Option zero must be specified, and must be last in the list of options. If no other options are specified, the following defaults apply:

- Physical units are inches.
- One curve is plotted, using a solid line 0.5 points wide. Points are connected by a Bézier curve that is not closed.
- XMIN, XMAX, YMIN and YMAX are determined automatically.
- All four borders are drawn, with automatically determined and labeled “tick” marks on the bottom and left borders.
- The output is L<sup>A</sup>TeX in file `splot.tex`.

10<sup>0</sup> 0 = Finish the curve, and the plot. Physical units are in inches.

1 = Finish the curve, and the plot. Physical units are in millimeters. 1 in. = 25.4 mm.

2 = Finish the curve, and the plot. Physical units are in points. 1 in. = 72.27pt.

3 = Finish this `mfpic` group, but allow the plot to be continued on a subsequent call. Use this for multiple curves if you have capacity problems with Metafont. This option requires that in your TeX or L<sup>A</sup>TeX document the file generated by SPLOT be included with a statement of the form `\hbox{\input ... }`, where ... is the file generated by SPLOT and there is a space preceding the

closing “}”. In L<sup>A</sup>T<sub>E</sub>X one can use mbox in place of hbox. (Units are needed only when plot finishes.)

- 4 = Finish the curve but allow the plot to be continued on a subsequent call.
  - 5 = More data for the current curve may be provided on a subsequent call.
- See special considerations in Section C.
- 10<sup>1</sup> 0 = Interpolate between points using a Bézier curve; do not close the curve.
  - 1 = Interpolate between points using a Bézier curve; close the curve with a Bézier curve.
  - 2 = Interpolate between points using a Bézier curve; close the curve with a straight line.
  - 3 = Connect points using straight lines; do not close the curve.
  - 4 = Connect points using straight lines; close the curve with a straight line.
  - 5 = Do not connect the points. This is frequently used with option 10.
- 10<sup>2</sup> Specifies how X is used to determine the abscissa,  $\xi$  of the point plotted, and how a border or axis with labels depending on X is labeled. When 10 <sup>$\xi$</sup>  is output for a label, if the major ticks are 1 unit apart in  $\xi$ , then minor ticks (if any) will be spaced logarithmically.

Value	Plot	Labels
0	$\xi = X$	$\xi$
1	$\xi = X$	10 <sup><math>\xi</math></sup>
2	$\xi = \log_{10} X$	$\xi$
3	$\xi = \log_{10} X$	10 <sup><math>\xi</math></sup> = X
4	Polar coordinates, $r = X \geq 0$	

Options for logarithmic or polar conversion apply to curves, and to options 14, 15, and 16. Those for logarithmic conversions apply to options 19–21. **Polar coordinates are not implemented in the current code.**

In the case of polar coordinates, the points plotted are  $(X \cos Y, X \sin Y)$ , and

- XMIN, XMAX, YMIN and YMAX are interpreted as  $r_{\min}$ ,  $r_{\max}$ ,  $\theta_{\min}$ , and  $\theta_{\max}$ , respectively.
- $r_{\min} \geq 0$  is required.
- Physical coordinates are Cartesian. The origin of the physical coordinate system coincides with the origin of the user (polar) coordinate system, and is within the figure.
- In a user Cartesian coordinate system derived from the polar coordinate system,  $x_{\min}$  is  $-r_{\max}$  if  $\theta_{\min} \leq 180^\circ + k \times 360^\circ \leq \theta_{\max}$  for some integer  $k$ , else it is

$\min(0, r_{\max} \cos \theta_{\min}, r_{\max} \cos \theta_{\max})$ .  $x_{\max}$ ,  $y_{\min}$  and  $y_{\max}$  are defined similarly. Given a point  $(r, \theta)$  in the user’s polar coordinate system, in the physical Cartesian coordinate system,  $X_{\text{physical}} = \text{XSIZE} \frac{r \cos \theta}{x_{\max} - x_{\min}}$ , and  $Y_{\text{physical}} = \text{YSIZE} \frac{r \sin \theta}{y_{\max} - y_{\min}}$ .

- The border indices are  $1 \Rightarrow \theta = \theta_{\min}$ ,  $2 \Rightarrow r = r_{\min}$ ,  $3 \Rightarrow \theta = \theta_{\max}$  and  $4 \Rightarrow r = r_{\max}$ .

10<sup>3</sup> The preceding table applies with X replaced by Y and  $\xi$  replaced by  $\eta$  unless in polar coordinates. In the case of polar coordinates,

Value	$\theta$ is given	Labels
0	in radians	degrees
1	in degrees	degrees
2	in radians	radians
3	in degrees	radians

10<sup>4</sup> Let  $x_b$  denote a vertical line drawn from border to border at the 0 point on the bottom border, if for this border  $\text{XMIN} < 0 < \text{XMAX}$ . Similarly let  $x_t$  be a vertical line at 0 as defined by the top border,  $y_\ell$  a horizontal line relative to the 0 on the left border, and  $y_r$  a horizontal line relative to the 0 on the right border. In all cases if the 0 line isn’t strictly inside the borders, the line is not drawn.

- 0 = Draw  $x_b$  and  $y_\ell$ .
- 1 = Do not draw any extra lines.
- 2 = Draw  $x_b$  only.
- 3 = Draw  $y_\ell$  only.
- 4 = Draw  $x_t$  only.
- 5 = Draw  $y_r$  only.
- 6 = Draw  $x_b$  and  $y_r$ .
- 7 = Draw  $x_t$  and  $y_\ell$ .
- 8 = Draw  $x_t$  and  $y_r$ .

10<sup>5</sup> 0 gives L<sup>A</sup>T<sub>E</sub>X output, 1 gives T<sub>E</sub>X output.

- 2 (Arguments K2 = NX, L2 = 1) L2 sets of ordinates are provided in Y. K2 is a whole number that gives NDIMY, the declared first dimension for Y ( $K2 \geq NX$ ), and L2 is a whole number that gives NY, the number of sets of ordinates, *i.e.* the effective second dimension for Y. The total dimension for Y must be  $\geq \text{NDIMY} \times \text{NY}$ . If a curve has been started but not yet finished, that is, SPLOT was most recently previously called with option 1, and with digit 10<sup>0</sup> of N1 = 5, and NY is changed by specifying this option, an error message is produced. NY must not be greater than the parameter MAXNY in SPLOT, which is currently 50.
- 3 (Argument P3 = 50) S3 specifies whether lines drawn are solid, dashed, or dotted, and the pen width for

solid or dashed lines. In the case of dashed lines one can indicate the length of the lines and the nominal space between them. In the case of dotted lines, one can specify the diameter of the dots and the space between them. Digits of S3 are used as follows:

10<sup>0</sup> Pen type.

= 0 Solid lines. (The only other thing to specify for this case is line width in 10<sup>1:2</sup> below.)

= 1 Dashed lines.

= 2 Dotted lines.

= 3,4 As for 1,2, except units for the length of dashes or the diameter of the dots is in deci-points instead of in points. See 10<sup>3:4</sup> below.

= 5–8 As for 1–4, except units for the length of the spaces, see 10<sup>5</sup> below, are in deci-points instead of in points.

10<sup>1:2</sup> The width for solid or dashed lines in deci-points. A value of 0 gives the default value of 5 deci-points. Here are lines of widths 3, 5, 7 and 10 deci-points: ||||.

10<sup>3:4</sup> The length of the dashed lines or the diameter of the dots. Units are points or deci-points, depending on the value of the low order digit of P3.

10<sup>5</sup> The nominal space between the dashed lines or the dots. Units are points or deci-points, depending on the value of the low order digit of P3.

The following table gives vertical lines showing various possibilities with both the spacing and the length for dashes and dots given in points.

Spacing:	1	1	2	2	2	3	3	3	3	4	4	4	4	4
Dashes:	1	2	1	2	3	1	2	3	4	1	3	4	5	7
Dots:	1	$\frac{3}{2}$	1	$\frac{3}{2}$	2	1	2	$\frac{5}{2}$	3	2	$\frac{5}{2}$	3	$\frac{7}{2}$	4



4 (Arguments N4, S4, T4 = 0) N4 is a whole number in which every decimal digit  $k$  for which  $1 \leq k \leq 6$  stipulates that S4 and T4 specify characteristics for border  $k$ . Border indices are: 1  $\Rightarrow$  bottom, 2  $\Rightarrow$  left, 3  $\Rightarrow$  top, 4  $\Rightarrow$  right, 5  $\Rightarrow$  X-axis if XMIN < 0 < XMAX, 6  $\Rightarrow$  Y-axis if YMIN < 0 < YMAX.

S4 is a whole number in which the decimal digits have the following meanings:

10<sup>0</sup> 0 = Border  $k$  is not drawn, else it is. If border  $k$  is not drawn, “tick” marks are neither drawn nor labeled.

1 = Major and minor “tick” marks are neither drawn nor labelled.

2 = No labels printed at major “tick” marks.

3 = Labels printed at major “tick” marks, except those at the ends of the border.

4 = Labels printed at major “tick” marks except those at the left (bottom) end of the border.

5 = Labels printed at major “tick” marks except those at the right (top) end of the border.

6 = Labels printed at all major “tick” marks.

Labels are never printed at minor “tick” marks, and will be skipped at some major tick marks if it appears the labels would overlap.

10<sup>1</sup> Length, in points, of arrow head to be drawn at the right or top end of the border. If 0, there is no arrow head.

10<sup>2:3</sup> Points plotted are not allowed to get closer to the border than this number of points.

10<sup>4:</sup> Gives the space in points to set aside for labels + outward pointing tick marks + border captions. If this is 0, this space will be estimated.

The defaults for bottom and left borders are 6, for top and right borders are 1, and for axes, are 0.

T4 is a whole number in which decimal digits have the following meanings:

10<sup>0:1</sup> The number of minor intervals within each major interval. If zero the value is determined automatically.

10<sup>2</sup> Defines how range on variable is expanded.

0 = Expand the range so that a major “tick” mark appears at automatically-determined ends of border  $k$  if the major “tick” mark would be at zero, else expand the range to a minor “tick” mark.

1 = Expand the range so that a major or minor “tick” mark appears at automatically-determined ends of border  $k$ .

2 = Expand the range so that a major “tick” mark appears at automatically-determined ends of border  $k$ .

else Do not expand the range.

10<sup>3:</sup> If nonzero, this is the index in COPT of the same kind of text allowed after a textual option character of 1–6 as defined in Section B.4. If more than one border/axis is indicated, the caption pointed to will be applied to all of them. Thus it is best to use COPT options 1-6 as described in Section B.4 below, unless this is being used for an extra data set, see Option 7.

**5** (Arguments N5 = 123456, A5 = 4.0pt, B5 = 3.0pt) N5 has the same meaning as N4. |A5| is the physical length of major “tick” marks along border  $k$ , and |B5| is the physical length of minor “tick” marks, both in points. “Tick” marks of positive length are directed inward, while “tick” marks of negative length are directed outward. If the physical length is  $>$  than the length of the plot ( $10^5$  should do this), the line is drawn the full length of the plot.

**6** (Arguments N6, X6, D6) N6 has the same meaning as N4. If  $D6 > 0$  then major “tick” marks will be drawn at all positions for which  $[XY]_{\text{MIN}} \leq X6 + k \times D6 \leq [XY]_{\text{MAX}}$  ( $k$  integer). If  $D6 > 0$  this option supersedes digits  $10^{0:1}$  of T4.

**7** (Argument B7, S7, T7, P7, U7) One may specify multiple data sets, and thus plot different curves on the same figure with different scalings and with a specified alignment. One should not specify this option without having already provided all of the data and options that reference a prior data set.

B7 This gives the index of the border used for annotations and labels for the new data set. See option 4 for definitions of borders 1–4. Values of 5 and 6 refer to the current data set for X and Y, and can be used to set values for P7 and U7 on the initial data set. If B7 indexes a border already in use by another data set, that border with associated tick marks and labels (if any) is output immediately, and the working area of the plot is reduced by moving the new location of this border towards the center of the plot.

S7 As for S4 for the new border,  $1 \leq B7 \leq 4$ .

T7 As for T4 for the new border,  $1 \leq B7 \leq 4$ .

P7 A distance in the same physical units as XSIZE and YSIZE from the left or bottom end of the border. See U7 below. If this is less than 0, there will be no effort to align the points for the different data sets.

U7 A value in user coordinates associated with this border. Data will be plotted in such a way that a data point with a coordinate with this value will be plotted at the distance from the left or bottom end of the border indicated by P7. This provides a means to align plots for the different data sets.

**8** (Arguments A8 = 0, B8 = 0) If  $A8 < B8$ , A8 (B8) gives the smallest (largest) value of X for the current dataset. Values outside this range will be *clipped*, *i.e.* will not appear in the plot. If all values are in this range, the plotting region is chosen as if these extreme values occurred.

**9** (Arguments A9 = 0, B9 = 0) Similar to option 8, but for Y instead of X.

**10** (Argument N10 ...) Plot symbols as specified by N10 at the current data points. In addition data points are connected or not as set by digit  $10^1$  of N1.

$N10 < 0$  means another N10 follows immediately, and  $N10 = 0$  draws no symbols. If several curves are being plotted (see option 2), the first uses the first |N10|, etc. If NY specifies more curves than the number of symbols specified, the specification for the last  $N10 \geq 0$  is re-used.

|N10| is a whole number in which decimal digits have the following meanings:

$10^0 \neq 1$  Plot symbols with vertices evenly spaced on a circle.

$10^0$  The number of vertices,  $v$ .  $v = 0$  describes a circle, in which case digits  $10^{1:2}$  are ignored.

$10^1$  The number of vertices,  $s$  to “skip” when drawing edges, except that  $s \geq v$  means “draw lines from the center to the vertices.” To include all the vertices when  $s < v$ ,  $\gcd(v, s + 1)$  polygons are drawn, each time starting one vertex counterclockwise from the previous starting vertex.

$10^2$  Rotation,  $n$ . The first vertex is at an angle of  $n \frac{45^\circ}{v}$  counterclockwise from the  $x$ -axis direction.

$10^3$  The width of line to use,  $w$ , in deci-points. If  $w = 0$  then  $w = 3$  is used. If  $w = 9$ , the symbol is filled.

$10^{4:5}$  The diameter of the circle in points. If 0, 6 is used.

$10^6$ : If this is not 0, the symbol is opaque *i.e.* pixels inside the symbol are cleared before drawing the symbol.

Possible values for |N10| include

603 $\triangle$	404 $\square$	215 $\star$	9215 $\blackstar$
203 $\nabla$	4 $\diamond$	255 $\star$	9603 $\blacklozenge$
3 $\triangleright$	14 $+$	426 $\ast$	9004 $\blacktriangle$
403 $\triangleleft$	414 $\times$	0 $\circ$	9000 $\bullet$

If  $s < v$  and  $v$  is odd, the given data are at the averages of the minima and maxima of the abscissas and ordinates of the vertices. Otherwise the given data are at the centers of circumcircles.

$10^{1:0}=1$  Plot error bars. NY must be a multiple of 2. Each point on a curve is specified by 3 numbers, *e.g.*  $x_i$ ,  $y_{i,1}$ , and  $y_{i,2}$  for the first curve. At each  $x_i$  a solid vertical line is drawn from  $y_{i,1} - y_{i,2}$  to  $y_{i,1} + y_{i,2}$ . The next three columns

of  $y$  would be used for the next curve, etc. Only the points  $(x_i, y_{i,1})$  are connected for the first curve, and similarly for later curves.

$10^2$  Length in points of a horizontal line drawn centered on the vertical line at the top and bottom of each error bar.

$10^3$  As for  $10^2$ , except used for a line at  $y_{i,1}$

$10^4$  Width in deci-points for the cross hatch lines.

$10^5$  The width of the vertical line,  $w$ , in deci-points. If  $w = 0$  then  $w = 3$  is used.

$10^{1:0} = 11$  As for  $10^{1:0} = 1$ , except each curve is specified by 4 numbers (NY must be a multiple of 3) and the top of the vertical line is  $y_{i,1} + y_{i,3}$  for the first curve.

$10^{1:0} = 21$  Draw “vector fields.” NY (see option 2) must be a multiple of 3. An arrow is drawn from  $(x_i, y_{i,1})$  to  $(x_i + y_{i,2}, y_{i,1} + y_{i,3})$  for the first curve. The next three columns of  $y$  would be used for the next curve, etc.

$10^2$  Length of the arrow head in points. If 0, no arrow head is drawn.

$10^3$  Size of circle in points to be drawn at  $(x_i, y_{i,1})$ . Set to 0 if no circle is desired.

$10^4$  Width in deci-points for the line used to draw the circle above. If this is 9, the circle is filled.

$10^5$  The width of the line use to draw the arrow,  $w$ , in deci-points. If  $w = 0$  then  $w = 3$  is used.

**11** (Arguments X11, Y11, N11...) Plot a single symbol as specified by N11 at (X11, Y11). |N11| has the same meaning as |N10|. If digit  $10^0$  of |N10| is 1, there are extra arguments corresponding to the values of  $y_{i,2}$  ... required by option 10.  $N11 \geq 0$  means (X11, Y11) (and any additional arguments) are in user coordinates, while  $N11 < 0$  means (X11, Y11) and additional arguments, if any, are in physical coordinates.

**12** (Argument S12 = 0) If S12  $\neq 0$  draw an arrow head, with a length of S12 points, at the end of the next open curve, or at the last point given in the next closed curve.

**13** (Arguments A13 = 100.0, B13 = 70.0, C13 = 50.0, D13 = 60.0, E13 = 30.0) Specify “pens” (as is done for option 3) for various kinds of lines. Values  $\leq 0$  select the default.

A13 For borders.

B13 For major “tick” marks.

C13 For minor “tick” marks.

D13 For lines drawn at  $X = 0$  or  $Y = 0$ .

E13 For rectangles or ellipses (see options 20, 21, 22, and 24)

**14** (Arguments X14, Y14, T14) Place a text annotation at (X14, Y14). The text begins at COPT(T14 / 10). If (T14 / 10) = 0, text starts immediately after the text for the last option of this type, or if this is the first, with the first position in COPT where this kind of option could appear. The text pointed to must be in the same form as the text following an option character of 1–6 as described in Section B.4, except the “{...}” contains text to be printed rather than a caption and “[...]” describes the justification relative to (X14, Y14).

T14 mod 10 has the following meaning:

0 = X14 and Y14 are in the current user coordinate system.

1 = X14 and Y14 are in physical coordinates.

2–3 as for 0–1 except don’t use the default prefix and postfix, any data of this type required is in the text string pointed to.

4–7 as for 0–3, and in addition an opaque borderless unfilled rectangle is placed “below” the annotation, but “above” any plotted curves or symbols.

**15** (Arguments V15, X15, Y15, T15). Place a number given by V15 on the plot at (X15, Y15). T15 is interpreted as for T14 except that if the value of T15 / 10 is 0, the default formatting is used. Otherwise the text pointed to in COPT has the same form as described in Section B.4 for text following an option character of “N”.

**16** (Arguments X16, T16) Place an annotation and/or a line or tick mark at the value given by X16 on one of the borders or axes. If X16 is not an interior point along this border or axis, this has no action. Digits of T16 are interpreted as follows.

$10^0$  Defines the type of line to draw.

= 0 No line.

= 1 A major tick mark.

= 2 A minor tick mark.

= 3 A solid line from border to border.

= 4 A dashed line from border to border with 3 point dashes and a 2 point space between them.

$10^1$  Gives the index of the border or axes to which this applies.

$10^2$ : Points to text in COPT for an annotation that is formatted with the same kind of rule as used for a numeric label, see Option 15. If 0 there is no annotation.

**17** (Argument A17, Y17) Values of Y equal to Y17 are considered to be “bad data.” If A17 is zero, “bad” values of Y are simply skipped. If A17 > 0 and a curve is being plotted, it will be terminated at the previous “good” value of Y, and re-started at the next “good” value of Y. If A17 < 0 then testing for “bad data” is turned off. When plotting a closed curve, if this option has been selected and A17 > 0, it is taken to be zero.

**18** Not used.

**19** (Arguments XA19, YA19, XB19 YB19) Draw a line in user coordinates from (XA19, YA19) to (XB19, YB19) using the currently selected line style parameters (including the arrow head), and the scaling set by the most recent appearance of options 8 or 9.

**20** (Arguments A20, B20, C20, D20) Draw a rectangle specified by points (A20, B20) and (C20, D20), in user coordinates of any two diagonally opposite corners of the rectangle, after all curves are drawn, but before any annotations specified by options 11 or 14 are placed. The rectangle is filled as specified by option 25.

**21** (Arguments X21, Y21, A21, B21, R21) Draw an ellipse centered at (X21, Y21) with major axis A21 and minor axis B21, and with the major axis rotated R21 degrees counterclockwise from the  $+x$ -axis direction. All but R21 are in user coordinates. If logarithms of user coordinates are ordinarily taken, logarithms of X21 and Y21 are computed before rotation. The input values of A21 and B21 are used without taking logarithms.

**22** Like 19, but in physical coordinates. Logarithms are never applied to coordinates in this case.

**23** Like 20, but in physical coordinates. Logarithms are never applied to coordinates in this case.

**24** Like 21, but in physical coordinates. Logarithms are never applied to coordinates in this case.

**25** (Arguments F25=0, ...) Specify filling of closed curves, rectangles defined by option 20 or 23, or ellipses defined by option 21 or 24. Some cases require extra arguments that are denoted by A25, B25, and C25. One must provide exactly as many arguments as the number required. F25 is a whole number in which the decimal digits have the following meaning:

- 10<sup>0</sup> 0 Do not fill – leave transparent.
- 1 Fill with solid black.
- 2 Erase.

3 Shade with dots of size A25 and spacing B25.

4 Hatch with lines of thickness A25, spacing B25 and angle C25 (in degrees counterclockwise from horizontal).

10<sup>1</sup> 0 = specification applies to curves, 1 = specification applies to rectangles defined by option 20, else specification applies to ellipses defined by option 21.

10<sup>2</sup> 0 = specification applies to next curve, rectangle or ellipses only. Else it applies, until changed by option 25. One can specify as many as 3 fill patterns with shading or hatching and all will be applied.

A25 and B25 are in points, 72.27 points/inch.

**26** Not used.

**27** (Argument L27 = 0) Print debugging output:

L27≤0 No debugging output

L27>0 Option settings

L27>1 Scratch file contents, including data.

## B.4 Textual Option Descriptions

Data at the beginning of COPT() consists of strings headed by a single character that identifies the kind of string that follows. After the last of such strings, COPT() may contain character strings pointed to by various options. Strings pointed to must begin with “[”, “{”, or “(”.

Options specified entirely in COPT() consist of a single letter or number code followed by text associated with the option. All letters used in options may be in either upper or lower case. Option codes may be preceded by blanks. Where we write “[”, “(”, or “{” (no text inside), the following is to be understood.

[ ] Always optional. Contains two or four letters between the brackets that define how an item is centered relative to some nominal position. The first is either **t**, **c**, or **b** for top, center (vertically), and bottom. The second letter is either **l**, **c**, or **r**, for left, center (horizontally), and right. One may follow the second letter with an **s** which indicates the text is “stacked” (vertically), and a following letter that must be **l**, **c**, or **r**, to indicate how the text is justified horizontally inside the stack. Inside a stack, text enclosed in balanced “{...}” or “\$...\$” is kept on a single line.

() Always optional. Contains information between the parentheses on formatting numbers and on the size of text (in points). Items can appear in any order,

with no intervening spaces. Letters can be in either upper or lower case, and # is used to denote an integer which if not present causes the nominal value to be used. The following can be specified.

. Always print a decimal point.

**F#** Font size in points. (The only case that makes sense if “()” is being used in connection with text output. Thus “(F12)” would be used to indicate text or numbers that are in 12 point.) The default is 9 point.

**D#** Number of significant digits that must be printed.

**A#** Number of digits that are required after the decimal point.

**B#** Number of digits that are required before the decimal point,

**X#**  $0 < \# < 10$ , bias for selecting the exponent notation. If # is zero, it is replaced by 5. The exponent notation is used if there are  $9 - \#$  or more zeros that are serving as place holders, else the usual format is used. Note that with an input # of 9, there will always be at least 0 zeros, and exponent notation must be used.

**{}** Never optional, but can be replaced by a “#” to get a default value. Between the braces there is a text string containing at least one “#” that is not preceded by a “\”. When outputting a number or text, this string is copied to the output with the first such “#” replaced by the number or text being output. This provides a means to change font, size, etc. The default for L<sup>A</sup>T<sub>E</sub>X is “{\small #}” and for T<sub>E</sub>X is “{#}”.

## Option

### Format                      Option Description

**F{File\_name}** Specifies a file name for the output. If this option is not selected, it is as if “F{splot.tex}” were used.

**Q** End of the option list. The end of the option list may also be indicated by {, ( or [ appearing where the first letter of an option code is expected — probably text used by option 14, 15, or 16.

**A(){}**  Default specification for printing numbers on borders and axes. Should precede any B, T, L, R, X, or Y specifications.

**B(){}**  Specification for printing numbers on bottom border.

**T(){}**  Specification for printing numbers on top border.

**L(){}**  Specification for printing numbers on left border.

**R(){}**  Specification for printing numbers on right border.

**X(){}**  Specification for printing numbers on x-axis.

**Y(){}**  Specification for printing numbers on y-axis.

**N[()]{}**  Specification for printing numbers interior to the plot. The default for “[ ]” is [c1].

**W[()]{}**  Specification for printing words or other text somewhere on the plot. If (...) is used, only the “F” for font size applies. The default for “[ ]” is [c1].

**C(){}**  Default specifications for printing border/axis captions.

**I{Input\_file\_name}** Specifies the name of a file from which the X and Y data are to be read. This file is opened with **FORM = 'UNFORMATTED'** and **ACCESS = 'SEQUENTIAL'**, and will have NX records read with a sequence of statements of the form  
`read (IO) X, (Y(J), J=1,NY)`  
 where IO is the unit number, automatically selected, associated with this file, and NY is 1 or, if set, the value of K2 in option 2. If NX is sufficiently large data is read to the end of the file.

**M{Raw plot output}** The raw plot output is sent directly to the plot device driver, except that \ serves as an escape character (*i.e.* \x for any “x” is replaced by x).

**“1–6”[()]{}{Border/axis\_caption}** Specification for a caption to be printed on a border/axis, where a single digit from 1 to 6 is used as in Option 4. A c for vertical positioning on a vertical line or for horizontal positioning on a horizontal line causes the text to be centered with respect to that direction outside any labels that may be present. Defaults for “[ ]” are [bc], [c1], [tc], [cr], [cr] and [tc] for the bottom, left, top, right, x-axis, and y-axis respectively.

The # is not used in this case. When using L<sup>A</sup>T<sub>E</sub>X the default “\small” is not generated in this case if the first character following the { is a \.

Thus “1{Label bottom border}” will place text centered below the bottom border with a baseline at the edge of the figure. And the default for the x-axis centers text vertically just past the right end of the axis. If one were to use [bc] on the top border, the text would have a baseline at the top of the figure. If defaults are used, a caption for a vertical line that requires more than two character lengths unstacked than it would require if stacked, will be stacked by adding sc to the positioning information.



## B.5 Incorporating the Output Into a T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X Document

SPLOT generates output intended to be processed by the `mfpic` package of T<sub>E</sub>X or L<sup>A</sup>T<sub>E</sub>X macros. See Section D below for more details. Using L<sup>A</sup>T<sub>E</sub>X or PDFL<sup>A</sup>T<sub>E</sub>X one can generate either “.dvi” files or “.pdf” files. We describe both approaches, but recommend using PDFL<sup>A</sup>T<sub>E</sub>X as it gives documents that are easier to share and it is easier to get set up. Also for many of our plots we had to use our own version of `gftopk` as the standard version does not set aside enough memory for the character sizes that are generated for the plots.

The general form of a L<sup>A</sup>T<sub>E</sub>X document that incorporates plots generated by SPLOT is:

```
\documentclass... % for LaTeX2e
% or \documentstyle for older LaTeX versions
...
\usepackage[metapost]{mfpic}
...
\begin{document}
...
\opengraphsfile{<fontname>}
...
\input <plotname>
% TeX reads the SPLOT output from the file
% <plotname>.tex, default is splot.tex
...
\input <anotherplotname>
...
\closegraphsfile
...
\end{document}
```

The processing steps for this document illustrate usage:

- 1 Run the programs that invoke SPLOT:  
    `pl16-03`      (Creates the symbols for option 10)  
    `drspplot`     (See Section C)
- 2 Process the document:  
    `pdflatex ch16-03`
- 3 Process the font produced in step 2:  
    `mpost pl16-03`  
    The name `pl16-03` was mentioned in the command  
    “`\opengraphsfile`” in this document.
- 4 Process the document again, this time incorporating  
    the generated plots from step 3:  
    `pdflatex ch16-03`

Programs above may have different names and different usages on different systems. If you are generating “.dvi” files and use a system that automatically updates a “.dvi” display when running L<sup>A</sup>T<sub>E</sub>X, you will likely need

to close this window after generating a new graphic font in order for the new graphic font to be displayed correctly.

To get dvi output. Replace “`\usepackage[metapost]{mfpic}`” with

“`\usepackage[metafont]{mfpic}`”.

Run `latex` instead of `pdflatex`.

Replace “`mpost pl16-03`” with “`mf pl16-03`”

Run `gftopk pl16-03.*gf`

Run `latex` again.

## B.6 Modifications for Double Precision

For double precision usage, change the `REAL` type statement to `DOUBLE PRECISION`, and change the subroutine name to `DPLOT`. The default file name for output, used if option `F` in `COPT()` is not selected, is `dplot.tex`.

## C. Examples and Remarks

The example given is that used to generate the plot in Chapter 2.2.

If one produces a curve or set of related curves by setting the  $10^0$  digit of  $N1 = 5$  and calling SPLOT several times to supply data values, the value of  $NY$  must be the same on every call that contributes to the construction of the curve(s).

If one produces a plot by making several calls to SPLOT, using digit  $10^0$  of  $N1$  to indicate that the plot is or is not to be finished on the current call, the following considerations are important:

- The specifications whether “tick” marks are linearly or logarithmically spaced are those in effect when the plot is finished.
- The specifications whether to plot data as given, or to plot  $\log_{10} X$  or  $\log_{10} Y$ , are remembered from one invocation of SPLOT to the next, but, if changed, are separately observed.
- The specification to use a polar coordinate system is the one in effect when the plot is finished.
- The units and values of `XSIZE` and `YSIZE` are those in effect when the plot is finished.

All options are reset to their default values when SPLOT is first invoked, or when invoked after finishing a plot. When SPLOT is invoked after having been invoked with instructions *not* to finish the plot, all optional settings retain their values from the previous invocation until changed by option selections in the current invocation, except that digit  $10^0$  of  $N1$  is reset to zero, and `L27` is reset to zero before the option selections are processed.

We also give a simple example here of taking output in a file to make a plot, and give scripts illustrating how to

get the plots. Program `plotf.f` takes output from a file and calls `dplot`. Scripts `plotpdf` and `plotdvi` shows how to do the whole job where the program `tedxr8` is the program used to generate the results and the  $\LaTeX$  file is `stepsel.tex`.

#### **plot.f**

```
c Plot data from a file; in this case file
c plot.out, assumed to have floating point
c numbers x_i, y_i, z_i on each line. y(x)
c and z(x) are plotted. Output in this case
c goes to file stepselp.tex.
      double precision X(1), Y(1), OPT(8)
      character COPT*27
      data COPT/'I{plot.out}F{stepselp.tex}Q'/
c
      Set for 2 curves ...
      data OPT/ 0.D0, 2.D0, 100000.D0, 2.D0,
c
      For x in [0, 15]
1 8.D0, 0.D0, 15.D0, 0.D0/
      call DPLLOT(4.5D0, 3.D0, X, 100000,
1 Y, OPT, COPT)
      stop
      end
```

#### **plotpdf**

```
#!/bin/bash
export RUNDIR=/m/math77/ode/dxr8/new
$RUNDIR/tedxr8 <$RUNDIR/test.in
$RUNDIR/plotf
pdflatex stepsel
mpost stepselp
pdflatex stepsel
```

#### **plotdvi**

```
#!/bin/bash
export RUNDIR=/m/math77/ode/dxr8/new
$RUNDIR/tedxr8 <$RUNDIR/test.in
$RUNDIR/plotf
latex stepsel
mf stepselp
gftopk stepselp.*gf
latex stepsel
```

## **D. Functional Description**

### **D.1 Sequence of Events**

Each time SPLOT is invoked, the character options in COPT are processed, then the numeric options in OPT are processed, then the data are processed. Most numeric options, and all data, are stored on a scratch file, and processed when digit  $10^0$  of  $N1 \leq 2$ .

Numeric options are processed in the order they appear, and the setting of one may affect the interpretation of another. When data are processed, the most recently

previous setting of an option that affects the data or their interpretation is effective, even if that setting occurred during a previous invocation of SPLOT for the same plot.

The effects of options or fields of options on subsequent options, and data, are summarized in the table at the beginning of Section B.3. The column labelled “Scratch File” indicates whether the option, or some of its fields, are put onto the scratch file, along with the data.

The plot border, axes, tick marks, labels, and captions are output first. Then data on the scratch file is output in the following order. First all data for a given `mfpic` group is output before that for another. Within each such group, curves are output, then single symbols, then rectangles and ellipses, and finally text annotations.

### **D.2 Context of Usage – Interaction with $\LaTeX$ and $\TeX$**

Plots are rendered into marks on the page by converting them to characters of a font. The `mfpic` package instructs  $\TeX$  or  $\LaTeX$  to put commands to produce the font into the file `<fontname>.mf` specified by the `\opengraphsfile` command (see Section B.5).

The document is thus produced by invoking the  $\TeX$  or  $\LaTeX$  program to produce the file `<fontname>.mf`, processing `<fontname>.mf` by the `mf` program, and then invoking  $\TeX$  or  $\LaTeX$  again to account for the “font metrics” and to incorporate the font “characters” that implement the plots.

After processing by the `mf` program, it may also be necessary to use the `gftopk` program, depending on how the `.dvi` file is ultimately converted to a format for a specific printer.

Finally, the document is converted from “device independent” (`.dvi`) form to a form usable by a specific printer by a `dvi` program specific to that printer.

Consult documentation specific to your system for details of usage of  $\TeX$ ,  $\LaTeX$ , `mf`, `gftopk` and `dvi` programs.

A version of the `mfpic` macros accompanies the libraries.

Some implementations of `gftopk` have insufficient capacity to process output from `mf` resulting from using `mfpic`. A version of `gftopk` that has more capacity accompanies the libraries. You will need a C compiler to compile it for your system.

## **E. Error Procedures and Restrictions**

Although this software has been used to produce the wide variety of plots in the MATH77 documentation, time constraints have meant that a good number of the options have not been checked. It is expected that there

will be bugs in some of these unchecked options. These will be fixed as they arise. Any errors due to a bug should be obvious from an examination of the plot if the bug allows the code to get this far. The C versions of this code have been checked only on the simple demonstration drivers and thus are even more likely to suffer from bugs.

An error message is produced if OPT(1) is returned non-zero. All errors are processed by the error message processor described in Chapter 19.3. If the error messages provided do not clear up the problem, looking at `<plotname>.tex` may clarify the problem for you.

## F. Supporting Information

The source language is ANSI Fortran 77.

Entry	Required Files
<b>SPLOT</b>	MESS, SMESS, SPLOT, SPLOT0
<b>DPLOT</b>	DMESS, MESS, SPLOT, SPLOT0

Design and code by Fred T. Krogh and W. Van Snyder, JPL, December 1997. Special thanks to Thomas Leathrum for creating `mfpic` and to Geoffrey Tobin for making his latest version of `mfpic` available and for answering questions on its use.

---

### DRSPLOT

```

program DRSPLO
c>> 1998-01-15 Krogh Generate the plot used in ch02-02.tex
c—S replaces "?: DR?PLO, DR?PL1, dr?plot, ?PLOT, ?ERF, ?ERFC, ?ERFCE
  integer MAXPT
  real RMAXPT, XA, XB, XSIZE, YSIZE
  parameter (MAXPT=101, RMAXPT=101.E0)
  parameter (XA = 0.0E0, XB = 5.0E0)
  parameter (XSIZE=3.2E0, YSIZE=2.75E0)
  real DX, OPT(17), X(MAXPT), Y(MAXPT, 3)
  external SERF, SERFC, SERFCE
  real SERF, SERFC, SERFCE
  integer I
  character COPT*41
  real PLONY, PLOTXT
  parameter (PLONY=2.E0, PLOTXT=14.E0)
  data OPT / 0.E0, PLONY, RMAXPT, 3.E0, PLOTXT, 1.1E0, .8E0, 0.E0,
1 PLOTXT, 1.2E0, .4E0, 0.E0, PLOTXT, 1.3E0, .1E0, 0.E0, 0.E0 /
  data COPT /
1 'F{drspLOT.tex}{erf(x)}{erfCE(x)}{erfc(x)}' /
c
  DX = (XB - XA) / (MAXPT - 1)
  do 100 I = 1, MAXPT
    X(I) = real(I - 1) * DX
    Y(I,1) = SERF(X(I))
    Y(I,2) = SERFC(X(I))
    Y(I,3) = SERFCE(X(I))
100 continue
  call SPLOT(XSIZE, YSIZE, X, MAXPT, Y, OPT, COPT)
c
c Don't use code below this point. This code is included to make
c possible automated checks of the software in different environments.
c
  call DRSPLO
  stop
end

  subroutine DRSPLO
c Print .tex output for checkout purposes.
c++ CODE for ~.C. is active
  character BUF*80
  integer I

```

```

    open (unit=10, FILE='drspplot.tex', status='OLD')
10 BUF = ' '
    read(10, '(A)', END=40) BUF
    do 20 I = 80, 2, -1
        if (BUF(I:I) .ne. ' ') go to 30
20    continue
30    print '(1X,A)', BUF(1:I)
    go to 10
40 continue
    close(10)
c++ CODE for .C. is inactive
c%% char buf[81];
c%% FILE *iofil;
c%% iofil = fopen("drspplot.tex", "r");
c%% while (fscanf( iofil, "%[^\n]", buf ) != EOF){
c%%     printf("%s\n", buf);
c%%     if (fscanf( iofil, "%[\n]", buf) == EOF) break;}
c%% fclose( iofil );
c++ END
    return
end

```

## ODSPLOT

```

\begin{mpic}[ 1.0 ]{ -8.000}{ 223.264}{ -10.500}{ 188.242}
\pen{ 1.000pt}
\lines{( 0.000, 0.000),( 223.264, 0.000)}
\mfsrc{
pickup pencircle scaled 0.700pt;
for x= 8.000 step 44.653 until 231.364:
draw(x, 10.500)*pt..(x, 14.500)*pt;
endfor;}
\mfsrc{
pickup pencircle scaled 0.500pt;
for x= 8.000 step 11.163 until 231.364:
draw(x, 10.500)*pt..(x, 13.000)*pt;
endfor;}
\tlabel[bl]( 0.000, -10.500){\small 0}
\tlabel[bc]( 44.653, -10.500){\small 1}
\tlabel[bc]( 89.306, -10.500){\small 2}
\tlabel[bc]( 133.958, -10.500){\small 3}
\tlabel[bc]( 178.611, -10.500){\small 4}
\tlabel[br]( 223.264, -10.500){\small 5}
\lines{( 0.000, 0.000),( 0.000, 188.242)}
\mfsrc{
pickup pencircle scaled 0.700pt;
for x= 10.500 step 37.648 until 198.842:
draw( 8.000, x)*pt..( 12.000, x)*pt;
endfor;}
\mfsrc{
pickup pencircle scaled 0.500pt;
for x= 10.500 step 18.824 until 198.842:
draw( 8.000, x)*pt..( 10.500, x)*pt;
endfor;}
\tlabel[br]( -3.500, 0.000){\small 0}
\tlabel[cr]( -3.500, 37.648){\small .2}
\tlabel[cr]( -3.500, 75.297){\small .4}
\tlabel[cr]( -3.500, 112.945){\small .6}
\tlabel[cr]( -3.500, 150.594){\small .8}

```

```

\tlabel[tr]( -3.500, 188.242){\small 1}
\lines{( 0.000, 188.242),( 223.264, 188.242)}
\lines{( 223.264, 0.000),( 223.264, 188.242)}
\pen{ 0.500pt}
\curve{( 0.0000, 0.0000),( 4.4653, 21.1703),( 11.1632, 52.0164)
,( 15.6285, 71.4158),( 20.0938, 89.5059),( 22.3264, 97.9802)
,( 24.5590, 106.0414),( 26.7917, 113.6714),( 29.0243, 120.8572)
,( 31.2570, 127.5910),( 33.4896, 133.8697),( 35.7222, 139.6949)
,( 37.9549, 145.0725),( 40.1875, 150.0120),( 42.4202, 154.5265)
,( 44.6528, 158.6321),( 46.8854, 162.3471),( 49.1181, 165.6920)
,( 51.3507, 168.6886),( 53.5834, 171.3598),( 55.8160, 173.7290)
,( 58.0486, 175.8200),( 60.2813, 177.6561),( 62.5139, 179.2605)
,( 64.7466, 180.6554),( 66.9792, 181.8620),( 69.2118, 182.9007)
,( 71.4445, 183.7903),( 73.6771, 184.5483),( 75.9098, 185.1912)
,( 78.1424, 185.7335),( 80.3750, 186.1889),( 82.6077, 186.5692)
,( 84.8403, 186.8853),( 87.0730, 187.1468),( 89.3056, 187.3619)
,( 91.5382, 187.5381),( 93.7709, 187.6816),( 96.0035, 187.7980)
,( 98.2362, 187.8918),( 100.4688, 187.9671),( 102.7014, 188.0273)
,( 104.9341, 188.0751),( 107.1667, 188.1129),( 111.6320, 188.1659)
,( 116.0973, 188.1981),( 122.7952, 188.2236),( 129.4931, 188.2348)
,( 138.4237, 188.2403),( 149.5869, 188.2421),( 167.4480, 188.2425)
,( 196.4723, 188.2425),( 223.2640, 188.2425)}
\curve{( 0.0000, 188.2425),( 4.4653, 167.0722),( 11.1632, 136.2261)
,( 15.6285, 116.8267),( 20.0938, 98.7366),( 22.3264, 90.2623)
,( 24.5590, 82.2011),( 26.7917, 74.5711),( 29.0243, 67.3853)
,( 31.2570, 60.6515),( 33.4896, 54.3728),( 35.7222, 48.5476)
,( 37.9549, 43.1700),( 40.1875, 38.2305),( 42.4202, 33.7160)
,( 44.6528, 29.6104),( 46.8854, 25.8954),( 49.1181, 22.5505)
,( 51.3507, 19.5539),( 53.5834, 16.8827),( 55.8160, 14.5135)
,( 58.0486, 12.4225),( 60.2813, 10.5863),( 62.5139, 8.9820)
,( 64.7466, 7.5871),( 66.9792, 6.3805),( 69.2118, 5.3418)
,( 71.4445, 4.4522),( 73.6771, 3.6941),( 75.9098, 3.0513)
,( 78.1424, 2.5090),( 80.3750, 2.0536),( 82.6077, 1.6733)
,( 84.8403, 1.3571),( 87.0730, 1.0957),( 89.3056, 0.8805)
,( 91.5382, 0.7044),( 93.7709, 0.5609),( 96.0035, 0.4445)
,( 98.2362, 0.3507),( 100.4688, 0.2753),( 102.7014, 0.2152)
,( 104.9341, 0.1674),( 107.1667, 0.1296),( 111.6320, 0.0766)
,( 116.0973, 0.0444),( 122.7952, 0.0189),( 129.4931, 0.0077)
,( 138.4237, 0.0022),( 149.5869, 0.0004),( 167.4480, 0.0000)
,( 196.4723, 0.0000),( 223.2640, 0.0000)}
\curve{( 0.0000, 188.2425),( 2.2326, 178.0755),( 8.9306, 152.2919)
,( 11.1632, 145.0119),( 15.6285, 132.0514),( 17.8611, 126.2708)
,( 20.0938, 120.8991),( 22.3264, 115.8991),( 24.5590, 111.2376)
,( 26.7917, 106.8850),( 29.0243, 102.8145),( 31.2570, 99.0024)
,( 33.4896, 95.4272),( 35.7222, 92.0695),( 37.9549, 88.9119)
,( 40.1875, 85.9386),( 42.4202, 83.1355),( 44.6528, 80.4894)
,( 46.8854, 77.9887),( 49.1181, 75.6227),( 53.5834, 71.2568)
,( 58.0486, 67.3235),( 62.5139, 63.7659),( 66.9792, 60.5360)
,( 71.4445, 57.5934),( 75.9098, 54.9034),( 80.3750, 52.4368)
,( 84.8403, 50.1684),( 91.5382, 47.0905),( 98.2362, 44.3486)
,( 104.9341, 41.8930),( 111.6320, 39.6827),( 118.3299, 37.6842)
,( 125.0278, 35.8694),( 131.7258, 34.2150),( 140.6563, 32.2249)
,( 149.5869, 30.4466),( 158.5174, 28.8489),( 169.6806, 27.0672)
,( 180.8438, 25.4878),( 192.0070, 24.0788),( 205.4029, 22.5771)
,( 218.7987, 21.2486),( 223.2640, 20.8393)}
\tlabel[bl]( 49.118, 150.594){\small erf(x)}
\tlabel[bl]( 53.583, 75.297){\small erfce(x)}
\tlabel[bl]( 58.049, 18.824){\small erfc(x)}
\end{mpic}

```