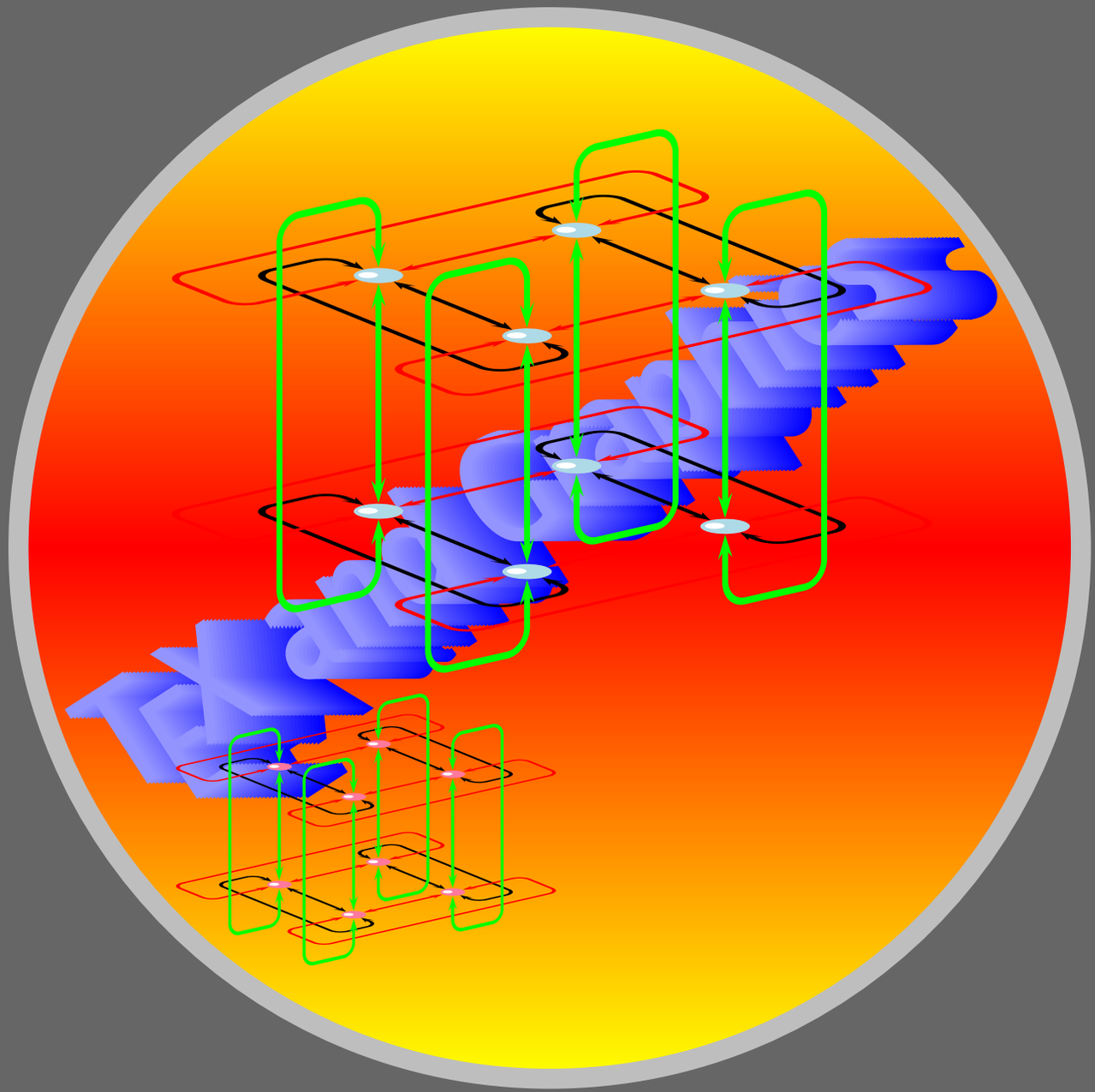


Plotting Tricks

Function plotting
Axes of coordinates
Data plotting

Online LaTeX Tutorial

Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to
tutorialteam@tug.org.in

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA

http://www.tug.org.in

7. Plotting Tricks

We have seen how we can join points to produce curves (for a mathematician, a *line* is also a curve!) with PSTricks, using such commands as `\psline`, `\pspolygon`, `\pscurve`, `\psccurve` and `\psecurve`. In all these commands, we'll have to *explicitly provide the coordinates* of the points joined to make the curve *within the command* itself. There are other methods of specifying the coordinates, either implicitly as a mathematical equation or as data piped in from another source. We discuss such tricks in this chapter.

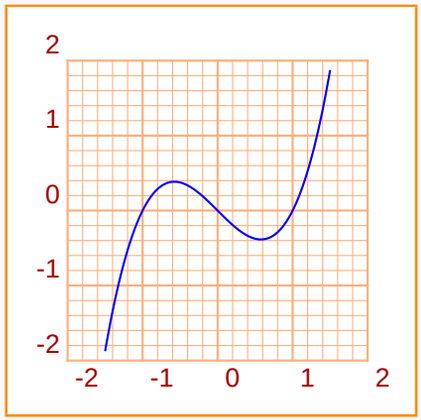
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to
tutorialteam@tug.org.in

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

7.1. Function plotting

For a mathematician, analytical geometry serves as a means of translation between algebra and geometry: an algebraic equation can be geometrically represented as the curve joining the points with coordinates satisfying the equation (called the *graph* of the equation) (and on the other hand, a curve can be algebraically represented as an equation specifying the relation between the coordinates of the points on it).

Thus for example, the graph of the equation $y = x^3 - x$ for $-1.5 \leq x \leq 1.5$ is as shown below:



The package pst-plot contains the command `\psplot` to draw curves specified by such equations. (Throughout the following, we assume that the package pst-plot is loaded by the command `\usepackage{pst-plot}` in the preamble.) The equation must be written in the PostScript Language. Thus the above equation is produced by

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-2,-3)(2,3)
  \psgrid[gridcolor=Apricot,%
          gridlabelcolor=Mahogany,%
          subgridcolor=Apricot]
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {-1.5}{1.5}{x 3 exp x sub}
 \end{pspicture}
```

The first command `\psgrid` just draws the coordinate grid, as we have seen earlier. The actual curve is drawn by the `\psplot` command. Let's look at this a bit more closely. The expression `x 3 exp x sub` is just the PostScirpt way of saying $x^3 - x$. In general, we must specify the equation which gives *the y-coordinate of each point of the curve in terms of the x-coordinate*. (or as mathematicians prefer to say, y should be given as a function of x). The numbers `-1.5` and `1.5` give respectively the minimum and maximum value of the x-coordinate in the curve to be drawn. Thus the basic drawing command is

`\psplot{`*min x*`}{`*man x*`}{`*equation*`}`

Looking back at our example, we see that here, the command `\psplot` also includes some parameters. The setting `linecolor=Blue` just makes the curve blue. What about `plotstyle=curve`? This makes `\psplot` draw a `\pscurve` through a certain number of points with coordinates computed from the equation. The number of points used is by default 50, but it can be controlled by the parameter `plotpoints`. This is illustrated in the next example:

Online LaTeX Tutorial

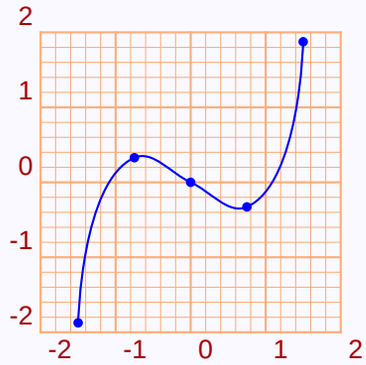
Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

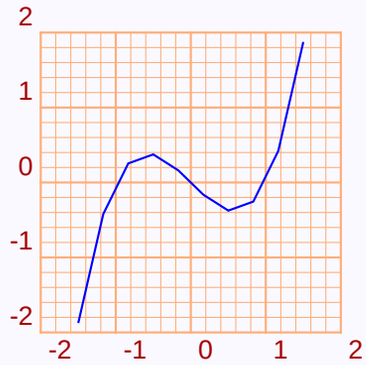
```
\begin{pspicture}(-2,-2)(2,2)
  \psgrid[gridcolor=Apricot,%
          gridlabelcolor=Mahogany,%
          subgridcolor=Apricot]
  \psplot[plotstyle=curve,%
          linecolor=Blue,%
          plotpoints=5,%
          showpoints=true]%
         {-1.5}{1.5}{x 3 exp x sub}
\end{pspicture}
```



The curve is not at all an accurate picture of the equation. In general, increasing the `plotpoints` increases the accuracy of the curve, but also increases the imaging time.

Coming back to the `plotstyle` parameter, we can also set it to `ccurve` or `ecurve` which draw a `\psccurve` (closed curve) or a `\psecurve` (suppressing the end-points in the picture). Other values are `line` and `polygon` which draw a `\psline` and a `\pspolygon` respectively. For example, with `plotstyle=line` the above graph is as shown below:

```
\begin{pspicture}(-2,-2)(2,2)
  \psgrid[gridcolor=Apricot,%
          gridlabelcolor=Mahogany,%
          subgridcolor=Apricot]
  \psplot[plotstyle=line,%
          linecolor=Blue,%
          plotpoints=10]%
         {-1.5}{1.5}{x 3 exp x sub}
\end{pspicture}
```



Online LaTeX Tutorial
Part II – Graphics
PSTricks

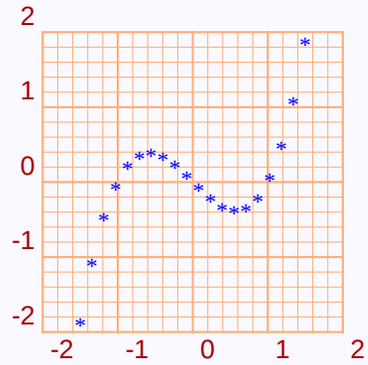
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

Note that with sufficiently many points specified in `plotpoints`, we get the same curve with `plotsyle=curve` and `plotstyle=line`, since PostScript draws any curve by joining short enough line segments).

There's one more value `plotstye=dots`, which draws only the dots used to draw the curve, without joining them. here, we can also use the various `dotstyle` values described in the first chapter. This is shown in the next example:

```
\begin{pspicture}(-2,-2)(2,2)
  \psgrid[gridcolor=Apricot,%
          gridlabelcolor=Mahogany,%
          subgridcolor=Apricot]
  \psplot[plotstyle=dots,%
          dotstyle=asterisk,%
          linecolor=Blue,%
          plotpoints=20]%
         {-1.5}{1.5}{x 3 exp x sub}
\end{pspicture}
```



Other relevant parameters can also be set for the `\psplot` command. Thus if we want to plot the function $f(x) = \sin x$ with x measured in radians (recall that PostScript computes $\sin x$ for x measured in degrees), we will have to scale down the x coordinates by $\frac{\dot{A}}{180} = 0.0174$. This can be done as shown below:

Online LaTeX Tutorial

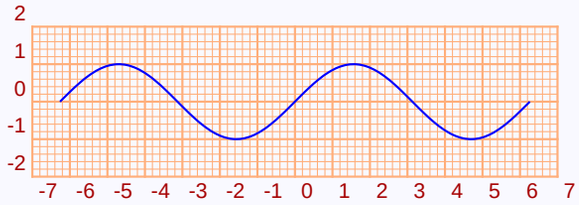
Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psset{unit=0.5}
\begin{pspicture}(-7,-2)(7,2)
  \psgrid[gridcolor=Apricot,%
          gridlabelcolor=Mahogany,%
          subgridcolor=Apricot,%
          gridlabels=8pt]
  \psplot[xunit=0.0174,%
          plotstyle=curve,%
          linecolor=Blue]%
         {-360}{360}{x sin}
\end{pspicture}
```



Sometimes, the relation between the x and y coordinates of points on a curve are not specified directly as an equation connecting x and y, but instead as an a pair of equations giving x and y in terms of a third parameter t. For example,

$$x = \frac{t}{1+t^2}$$
$$y = \frac{t^3}{1+t^2}$$

The graph of this function for $-3 \le t \le 3$ can be drawn as shown below:

Online LaTeX Tutorial

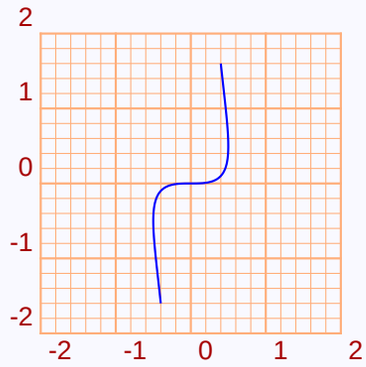
Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

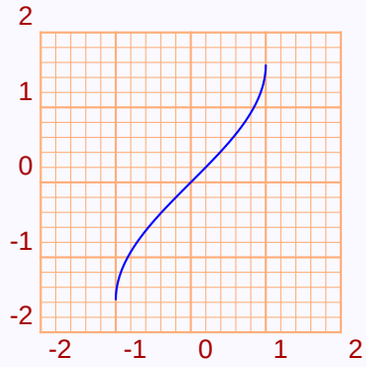
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-2,-2)(2,2)
  \psgrid[gridcolor=Apricot,%
                gridlabelcolor=Mahogany,%
                subgridcolor=Apricot]%
                (-2,-2)(2,2)
  \parametricplot[plotstyle=curve,%
                  linecolor=Blue]%
                  {-2}{2}%
                  {t t 2 exp 1 add div t 3 exp
                   t 2 exp 1 add div}
\end{pspicture}
```



The parametric plot provides an easy method to draw inverse functions. The example below shows how we can plot the inverse sine function:

```
\begin{pspicture}(-2,-2)(2,2)
  \colgrid
  \parametricplot[linecolor=Blue]%
                 {-1.5708}{1.5708}%
                 {57.2958 t mul sin t}
\end{pspicture}
```



Note the multiplication by $57.2958 = \frac{180}{\pi}$ for conversion to radians.

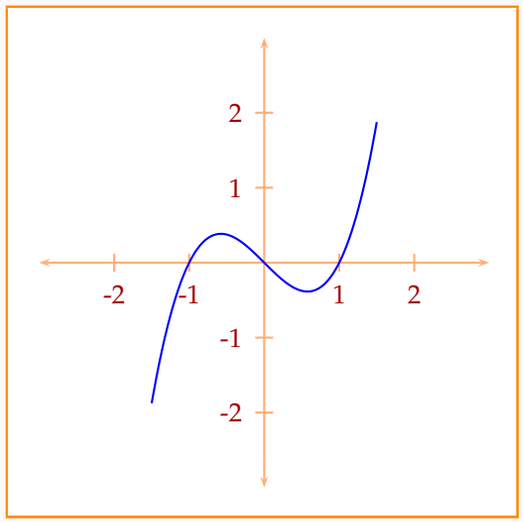
Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

7.2. Axes of coordinates

Often in mathematical documents, graphs of equations are shown, not with the complete coordinate grid, but only with the x and y axes as shown below:



We can draw such axes using `\psline`, but pst-plot also includes the command `\psaxes` to draw such axes more conveniently.

7.2.1. Drawing the axes

The command

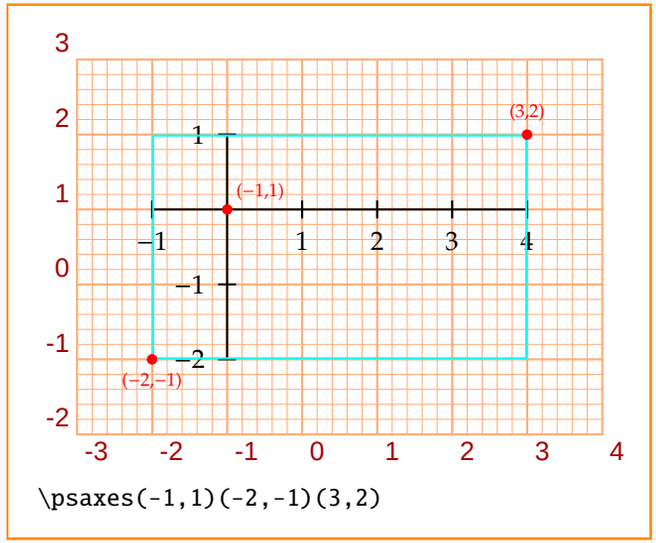
$$\texttt{\textbackslash psaxes}(x_0, y_0)\,(x_1, y_1)\,(x_2, y_2)$$

draws x and y axes with the x-axis from x_1 to x_2 and the y-axis from y_1 to y_2 with the origin (for the axes) at (x_0, y_0); that is, the axes are enclosed in an (imaginary) rectangle with opposite corners at (x_1, y_1) and (x_2, y_2) and the axes intersect at (x_0, y_0). This is illustrated in the picture below, where we have

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

shown the bounding rectangle in Cyan and the coordinates of the relevant points with respect to the current coordinate system in red:



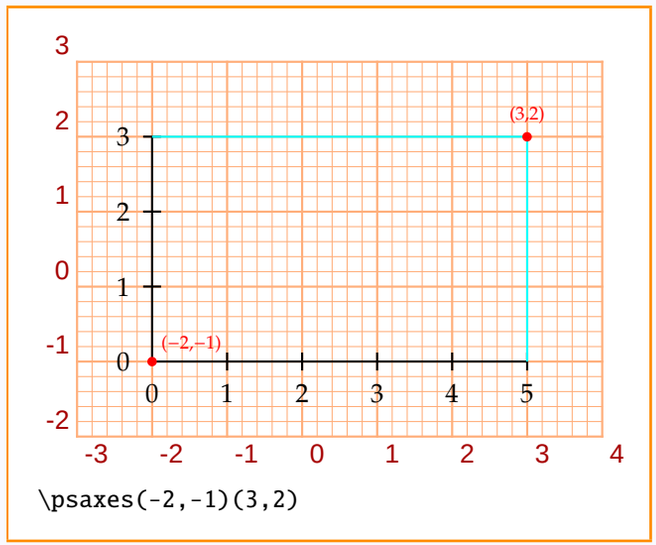
If the origin for the axes is to be the same as one of the corners of the bounding rectangle, we need not repeat it in the command: thus if we say

$$\texttt{\textbackslash psaxes}(x_1, y_1)(x_2, y_2)$$

then the origin and one corner of the bounding rectangle will be at (x_1, y_1) (and of course, the other corner will be at (x_2, y_2)), as shown below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in



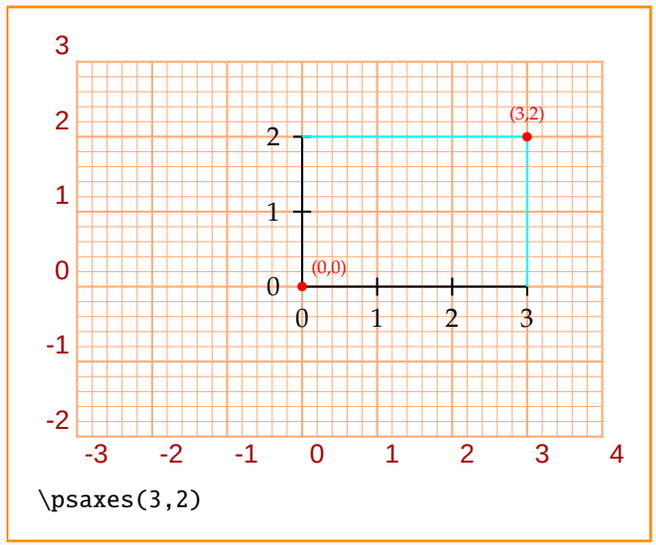
Thus \psaxes$(x_1, y_1)\,(x_2, y_2)$ is equivalent to \psaxes$(x_1, y_1)\,(x_1, y_1)\,(x_2, y_2)$. Again, if the origin for the axes is to be the same as one of the corners of the bounding rectangle and if this point is to be the same as the origin of the current coordinate system, then we need only specify only the other corner of the rectangle; that is,

\psaxes$(x, y)$

draws axes with the origin (of the axes) and one corner of the bounding rectangle at the origin of the coordinate system currently in effect, and the other corner of the rectangle at (x, y), as shown below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

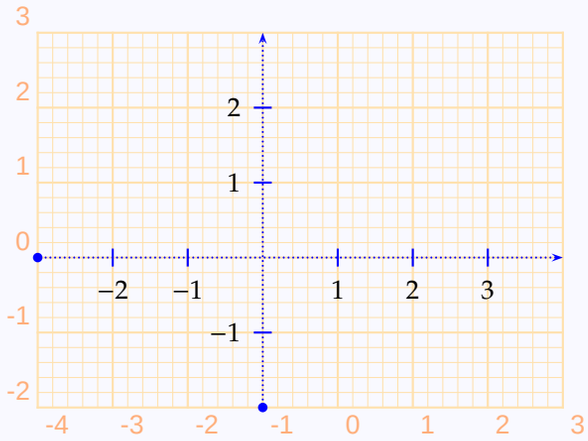
The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in



Thus `\psaxes(`x, y`)` is equivalent to `\psaxes(0,0)(0,0)(`x, y`)`.

We can set various parameters such as `linecolor` within the `\psaxes` command and also specify arrow terminations. The example below illustrates some of these.

```
\begin{pspicture}(-2,-3)(3,4)
  \psaxes[linecolor=Blue,%
          linestyle=dotted,
          dotsep=1pt]%
         {*->}%
         (-1,0)(-4,-2)(3,3)
  \end{pspicture}
```



Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

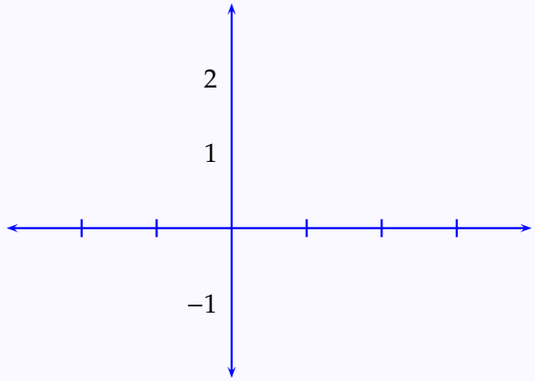
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

We have included a background grid for reference and it is not produced by the code given. Note that the first arrowhead * is used for the tips at $(-4, 0)$ and $(0, -2)$ and the second arrowhead > is used at $(3, 0)$ and $(0, 3)$.

7.2.2. Marking the axes

The `\psaxes` command not only draws the axes, but puts equally spaced marks on the axes and names them. This is controlled by the parameters `ticks` and `labels`. Each of these has four possible values, `all` (which puts ticks or labels on both the axes, and is the default value), `x` (ticks or labels on the x-axes alone, `y` (on y-axis alone) and `none` (on neither axes). This is illustrated in the example below:

```
\begin{pspicture}(-4,-2)(3,3)
  \psaxes[linecolor=Blue,%
          ticks=x,%
          labels=y]%
         {<->}%
         (-1,0)(-4,-2)(3,3)
  \end{pspicture}
```



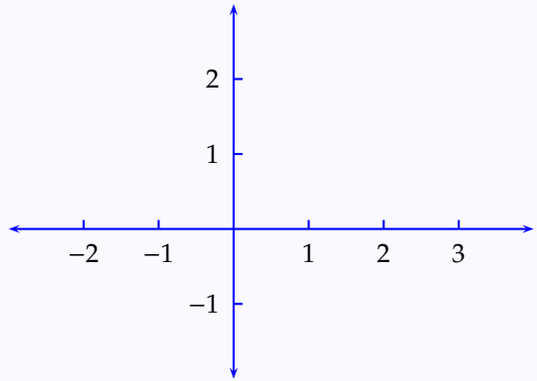
The position and size of the ticks can also be adjusted. Whether the ticks extend to either side of the origin or only to one side is determined by the parameter `tickstyle`. Its possible values are `full` (ticks extending to either side, which is the default), `top` (ticks extending only to the opposite side of the labels) and `bottom` (ticks on same side as the labels). The length of the

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

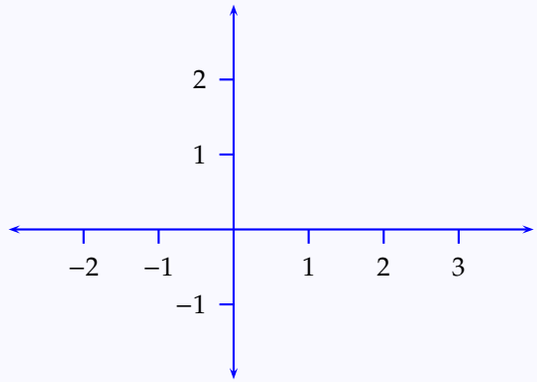
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

ticks is controlled by the `ticksize` parameter with default value 3 pt. These are illustrated in the next two examples:

```
\begin{pspicture}(-4,-2)(3,3)
  \psaxes[linecolor=Blue,%
          tickstyle=top]%
         {<->}%
         (-1,0)(-4,-2)(3,3)
  \end{pspicture}
```




```
\begin{pspicture}(-2,-3)(3,4)
  \psaxes[linecolor=Blue,%
          tickstyle=bottom,%
          ticksize=5pt]%
         {<->}%
         (-1,0)(-4,-2)(3,3)
  \end{pspicture}
```



In some of our examples above, the origin is labeled (of course as (0, 0)), but in some others, it is not. The default behavior is to label the origin, unless the label will fall on any of the axes. We can turn off the labeling of origin (even in those cases, where it would be otherwise done), by setting the parameter `showorigin` to `false`. (By default it is `true`). Look at the example below:

Online LaTeX Tutorial

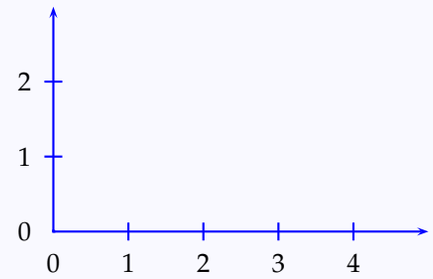
Part II – Graphics

PSTricks

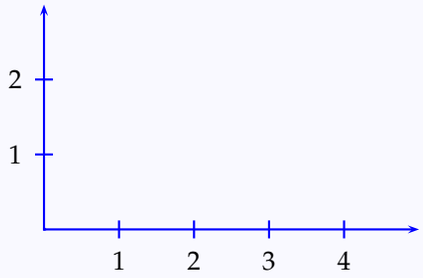
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-2,-2)(3,2)
  \psaxes[linecolor=Blue]%
         {<->}%
         (-2,-1)(3,2)
\end{pspicture}
```




```
\begin{pspicture}(-2,-2)(3,2)
  \psaxes[linecolor=Blue,%
          showorigin=false]%
         {<->}%
         (-2,-1)(3,2)
\end{pspicture}
```



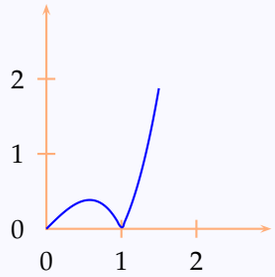
When the axes enclose a single quadrant of the plane, the labels are, by default, placed outside the quadrant (so as not to obstruct the graph within) as shown in the pictures below:

Online LaTeX Tutorial
Part II – Graphics
PSTricks

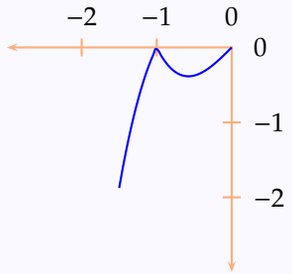
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,0)(3,3)
  \psaxes[linecolor=Apricot]%
         {<->}%
         (3,3)
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {0}{1.5}%
         {x 3 exp x sub abs}
  \end{pspicture}
```




```
\begin{pspicture}(0,0)(-3,-3)
  \psaxes[linecolor=Apricot]%
         {<->}%
         (-3,-3)
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {-1.5}{0}%
         {x 3 exp x sub abs neg}
\end{pspicture}
```



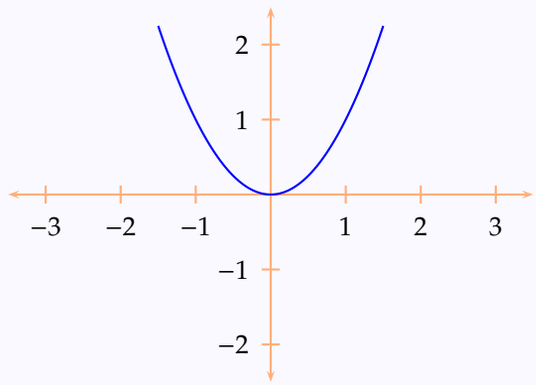
If the axes are specified by two corners lying in two different quadrants, the labels are marked *inside* the quadrant of the first corner specified. The examples below will make this clear:

Online LaTeX Tutorial
Part II – Graphics
PSTricks

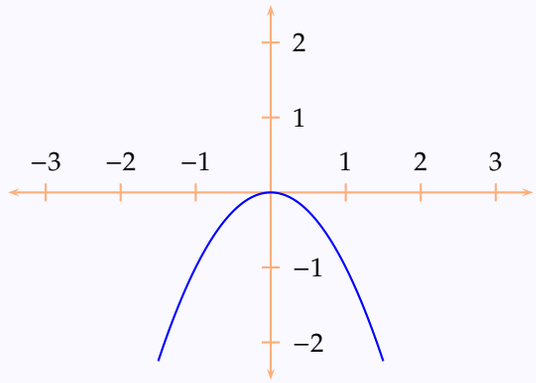
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-3,-3)(3,3)
  \psaxes[linecolor=Apricot]%
         {<->}%
         (0,0)(-3.5,-2.5)(3.5,2.5)
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {-1.5}{1.5}%
         {x 2 exp}
\end{pspicture}
```




```
\begin{pspicture}(-3,-3)(3,3)
  \psaxes[linecolor=Apricot]%
         {<->}%
         (0,0)(3.5,2.5)(-3.5,-2.5)
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {-1.5}{1.5}%
         {x 2 exp neg}
  \end{pspicture}
```



Plotting Tricks

Online LaTeX Tutorial

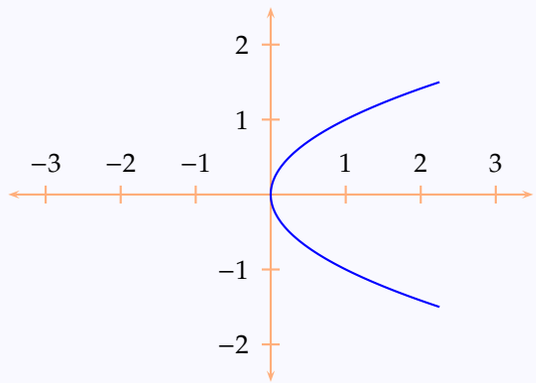
Part II – Graphics

PSTricks

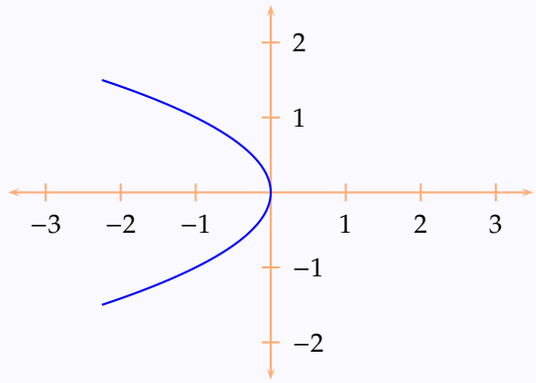
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-3,-3)(3,3)
  \psaxes[linecolor=Apricot]%
         {<->}%
         (0,0)(-3.5,2.5)(3.5,-2.5)
  \parametricplot[plotstyle=curve,%
                  linecolor=Blue]%
                 {-1.5}{1.5}%
                 {t 2 exp t}
 \end{pspicture}
```




```
\begin{pspicture}(-3,-3)(3,3)
  \psaxes[linecolor=Apricot]%
         {<->}%
         (0,0)(3.5,-2.5)(-3.5,2.5)
 \parametricplot[plotstyle=curve,%
                  linecolor=Blue]%
                 {-1.5}{1.5}%
                 {t 2 exp neg t}
 \end{pspicture}
```



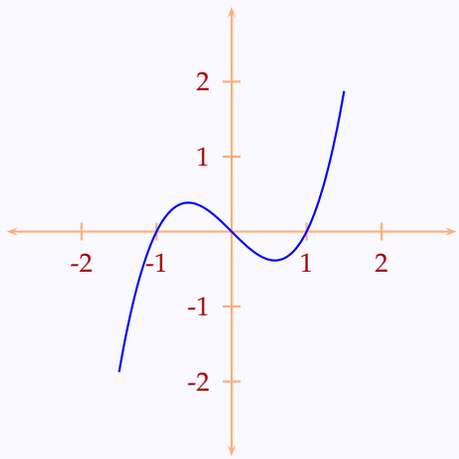
The distance between labels and axes can be controlled by the `labelsep` parameter. The style of the labels (such as font, color and so on) is controlled by the command `\pshlabel`, for horizontal (that is, along the x-axis) and `\psvlabel`, for the vertical (that is, along the y-axis). The complete code for the picture at the beginning of this section can be now given:

Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-3,-3)(3,3)
  \renewcommand{%
    \pshlabel}[1]{%
    \color{Mahogany}#1}
  \renewcommand{%
    \psvlabel}[1]{%
    \color{Mahogany}#1}
  \psaxes[linecolor=Apricot]%
         {<->}%
         (0,0)(-3,-3)(3,3)
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {-1.5}{1.5}%
         {x 3 exp x sub}
\end{pspicture}
```

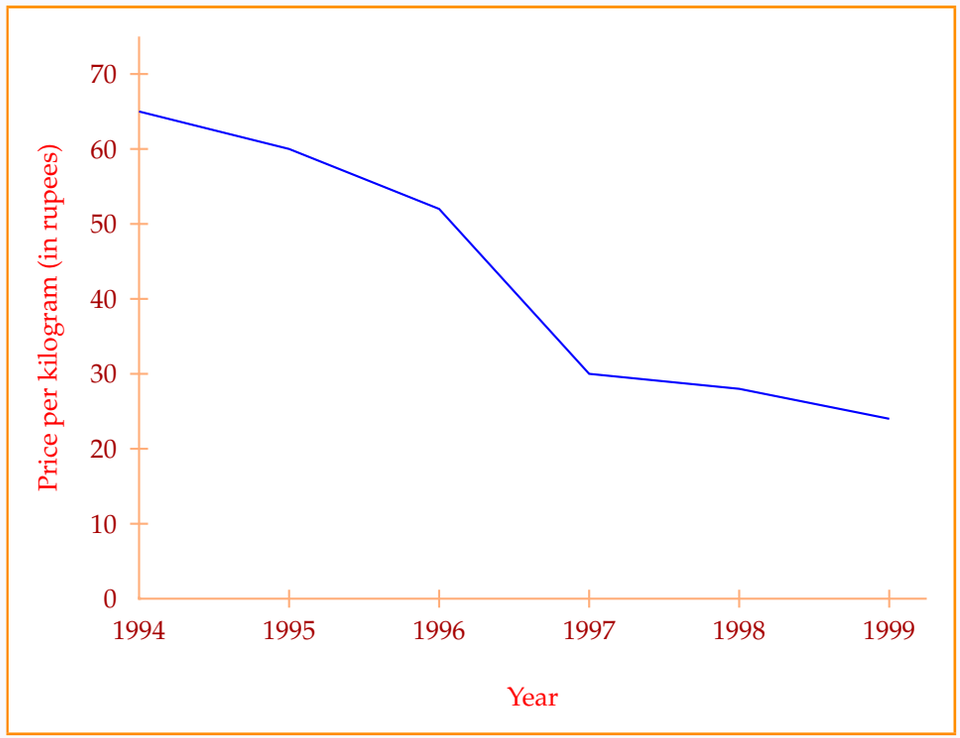


The command \psaxes, by default, produces equally spaced ticks on each axes at either side of the origin and labels them with consecutive integers. But sometimes, we do need axes marked differently as in the picture below:

Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in



Here, we need the origin to be marked with the numbers 1994 (on the x-axis) and 0 (on the y-axis), the x-axis to be marked with the numbers $1994, 1995, \ldots$ spaced at 2 cm and the y-axis marked with $0, 10, 20, \ldots$ spaced at 1 cm. All these can be done automatically, by setting certain parameters of the `\psaxes` command. These are shown in the table below:

parameter	meaning	default
Ox	Label at the origin on the x-axis	0
Oy	Label at the origin on the y-axis	0
Dx	increment in labels for the x-axis	1
Dy	increment in labels for the y-axis	1
dx	distance between ticks along the x-axis	`Dx×\psxunit`
dy	distance between ticks along the y-axis	`Dy×\psyunit`

Online LaTeX Tutorial
Part II – Graphics
PSTricks

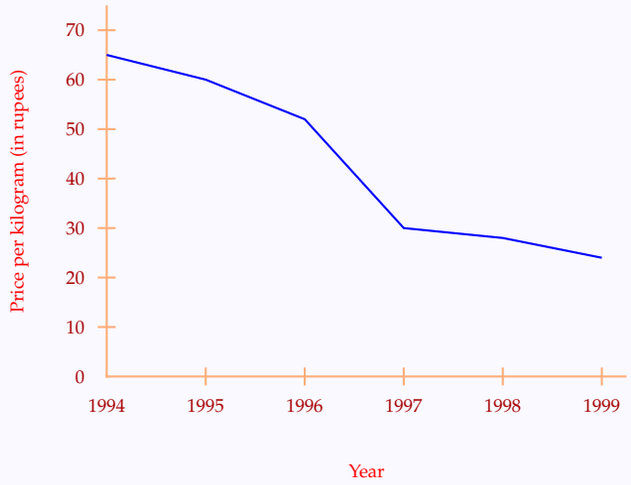
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to
tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

Note that the entries in the first column of the first row are "oh" (uppercase) and *not zero*. Also, `\psxunit` and `\psyunit` (in the last row) are TEX dimensions specifying the units along the x-axis and the y-axis.

Thus the picture above (actually, a scaled down version—to get the picture itself, simply remove the `\psset` command and all the `\scrtiptsize` commands) is produced as shown below:

```
\psset{unit=0.66}
 \renewcommand{%
   \pshlabel}[1]{%
    \scriptsize\color{Mahogany}#1}
 \renewcommand{%
  \psvlabel}[1]{%
   \scriptsize\color{Mahogany}#1}
 \begin{pspicture}(0,0)(10,7)
   \psaxes[linecolor=Apricot,%
           Ox=1994,Oy=0,%
           Dx=1,Dy=10,%
           dx=2,dy=1]%
          (10.5,7.5)
  \psline[linecolor=Blue]%
         (0,6.5)(2,6)(4,5.2)%
         (6,3)(8,2.8)(10,2.4)
  \uput[d](5.25,-1.5){%
    \scriptsize\color{Red} Year}
  \uput[l](-1.5,3.75){%
    \rput{90}{%
      \scriptsize\color{Red}%
       Price per kilogram (in rupees)}}
\end{pspicture}
```



These parameters can be set to non-integer values also. Thus the plot of $f(x) = \sin x$ with x measured in radians can be drawn as below:

Online LaTeX Tutorial

Part II – Graphics

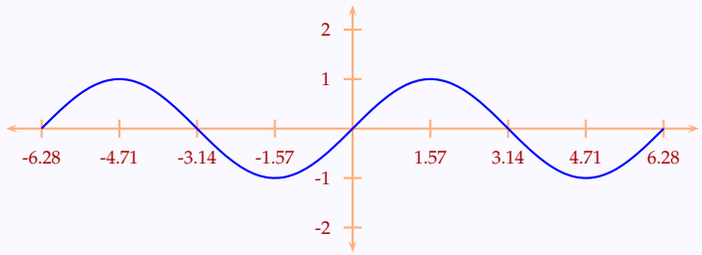
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA

http://www.tug.org.in

```
\psset{unit=0.66}
\renewcommand{%
  \pshlabel}[1]{%
    \scriptsize\color{Mahogany}#1}
renewcommand{%
  \psvlabel}[1]{%
    \scriptsize\color{Mahogany}#1}
\begin{pspicture}(-7,-2)(7,2)
  \psaxes[linecolor=Apricot,%
          Dx=1.57,dx=1.57]%
          {<->}%
          (0,0)(-7,-2.5)(7,2.5)
  \psplot[plotstyle=curve,%
          linecolor=Blue]%
         {-6.2832}{6.2832}%
         {57.2958 x mul sin}
\end{pspicture}
```



In using such non-integer values for these parameters, we must keep in mind that `0x` and `Dx` or `0y` and `Dy` must have the same number of digits to the right of the decimal point. However, even if `Dx` and `Dy` are not integers, `0x` and `0y` can be so.

Finally, using the `axesstyle` parameter, we can choose to have a frame instead of axes (if the axes enclose a single quadrant) or no axes at all (but with ticks and labels). The possible values of this parameter are `axes` (which is the default), `frame` or `none`. These possibilities are illustrated in the examples below:

Online LaTeX Tutorial

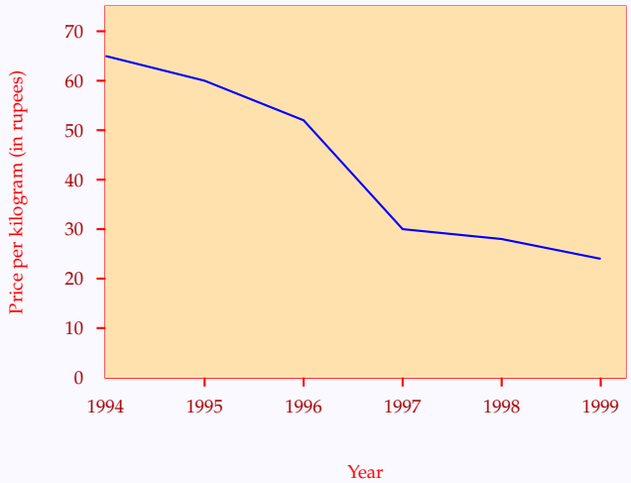
Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psset{unit=0.66}
 \renewcommand{%
   \pshlabel}[1]{%
    \scriptsize\color{Mahogany}#1}
 \renewcommand{%
  \psvlabel}[1]{%
   \scriptsize\color{Mahogany}#1}
\definecolor{LightApricot}%
  {cmyk}{0,0.27,0.47,0}
\begin{pspicture}(0,0)(10,7)
   \psaxes[axesstyle=frame,%
           fillstyle=solid,%
           fillcolor=PaleApricot,%
           linecolor=Apricot,%
           Ox=1994,Oy=0,%
           Dx=1,Dy=10,%
           dx=2,dy=1,
           tickstyle=bottom]%
          (10.5,7.5)
  \psline[linecolor=Blue]%
         (0,6.5)(2,6)(4,5.2)%
         (6,3)(8,2.8)(10,2.4)
  \uput[d](5.25,-1.5){%
    \scriptsize\color{Red} Year}
  \uput[l](-1.5,3.75){%
    \rput{90}{%
      \scriptsize\color{Red}%
       Price per kilogram (in rupees)}}
\end{pspicture}
```



Plotting Tricks

Online LaTeX Tutorial

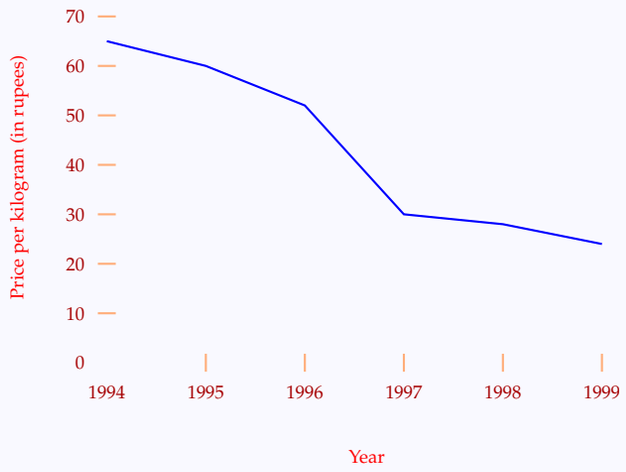
Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psset{unit=0.66}
 \renewcommand{%
   \pshlabel}[1]{%
    \scriptsize\color{Mahogany}#1}
 \renewcommand{%
  \psvlabel}[1]{%
   \scriptsize\color{Mahogany}#1}
 \begin{pspicture}(0,0)(10,7)
   \psaxes[axesstyle=none,%
           linecolor=Apricot,%
           Ox=1994,Oy=0,%
           Dx=1,Dy=10,%
           dx=2,dy=1]%
          (10.5,7.5)
  \psline[linecolor=Blue]%
         (0,6.5)(2,6)(4,5.2)%
         (6,3)(8,2.8)(10,2.4)
  \uput[d](5.25,-1.5){%
    \scriptsize\color{Red} Year}
  \uput[l](-1.5,3.75){%
    \rput{90}{%
      \scriptsize\color{Red}%
       Price per kilogram (in rupees)}}
\end{pspicture}
```



Online LaTeX Tutorial

Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

7.3. Data plotting

The command `\psplot` is used to plot a curve from its algebraic equation. As mentioned above, the coordinates of points to be plotted is computed by PostScript and a curve (depending on the `plotstyle` parameter) connecting these is drawn. If instead, the coordinates of the points are to be explicitly specified, we use the `\pscurve` or `\psline` command. But, if the set of coordinates is large or is generated by some other program, then it's tedious (and inelegant) to include all these in the TEX code. The pst-plot package has some plotting commands by which a list of coordinates contained in an external file can be piped into TEX code.

For example, consider the file `data1.csv` with contents as shown below, exported from the gnumeric spreadsheet program:

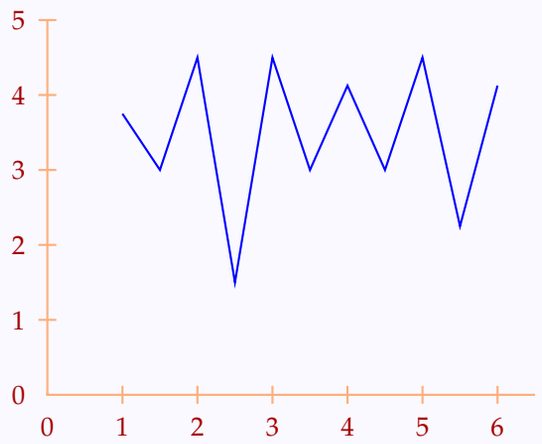
```
1,3.75
1.5,3
2,4.5
2.5,1.5
3,4.5
3.5,3
4,4.125
4.5,3
5,4.5
5.5,2.25
6,4.125
```

To plot a line graph of this data, we can use the command `\fileplot` as shown below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,0)(6.5,5)
  \renewcommand{%
    \pshlabel}[1]{%
       \color{Mahogany} #1}
  \renewcommand{%
    \psvlabel}[1]{%
      \color{Mahogany} #1}
  \psaxes[linecolor=Apricot]%
         (6.5,5)
  \fileplot[linecolor=Blue]%
           {data1.csv}
\end{pspicture}
```



In general, the command

\fileplot[*parameters*]{*filename*}

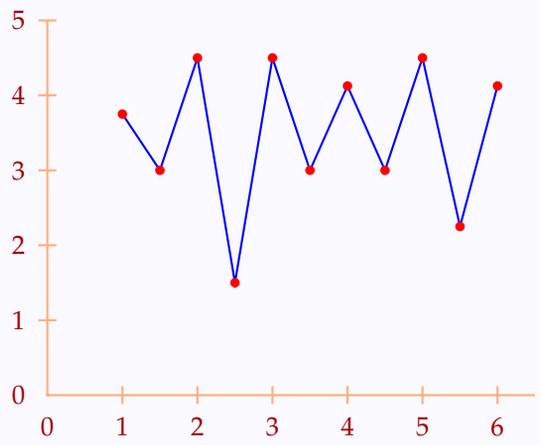
can be used to plot the graph connecting the points with coordinates listed in the file with name \textit{filename}, contained in the working directory. The file should only contain the coordinates and perhaps comments marked with % and nothing else. The numbers can be delimited by white space, commas, parentheses ( ) or curly braces, { }. (In the example above, both white spaces and commas are used).

The \fileplot command recognizes only the line, polygon and dots styles of plotting. Also, it ignores the parameters arrows, linearc and showpoints parameters. However , we can display the points plotted by including a second \fileplot with plotstyle=dots as shown below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,0)(6.5,5)
  \renewcommand{%
    \pshlabel}[1]{%
       \color{Mahogany} #1}
  \renewcommand{%
    \psvlabel}[1]{%
      \color{Mahogany} #1}
  \psaxes[linecolor=Apricot]%
         (6.5,5)
  \fileplot[linecolor=Blue]%
           {data1.csv}
  \fileplot[plotstyle=dots,%
            linecolor=Red]%
           {data1.csv}
\end{pspicture}
```



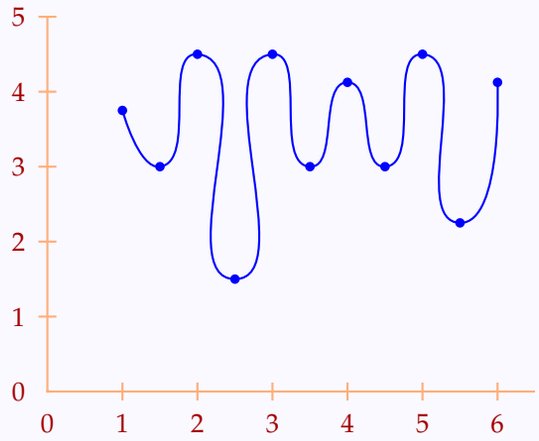
This example also illustrates an advantage of using external files to hold lists of coordinates, namely the re-usability of such lists.

Another command available for plotting data from external files is `\dataplot`. To use this, we must first name a command to read the data from the external file using the command `\readdata` and then use this new command as an argument to `\dataplot`. This is illustrated in the example below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,0)(6.5,5)
  \renewcommand{%
    \pshlabel}[1]{%
      \color{Mahogany} #1}
    \renewcommand{%
      \psvlabel}[1]{%
        \color{Mahogany} #1}
    \psaxes[linecolor=Apricot]%
           (6.5,5)
    \readdata{\picdata}{data1.csv}
    \dataplot[plotstyle=curve,%
              showpoints=true,%
             linecolor=Blue]%
             {\picdata}
\end{pspicture}
```



Note that the `\dataplot` command can accepts the `curve` style of plotting and also the `showpoints` parameter.

This command also has a facility to include coordinates directly (and then perhaps re-used) using the command `\savedata`. Thus in the last example, instead of `\readdata`...`\dataplot`, the code

```
\savedata{\grphdata}[%
    {1,3.75 1.5,3 2,4.5 2.5,1.5
     3,4.5 3.5,3 4,4.125 4.5,3
     5,4.5 5.5,2.25 6,4.125}]
\dataplot[plotstyle=curve,%
          showpoints=true,%
          linecolor=Blue]%
         {\grphdata}
```

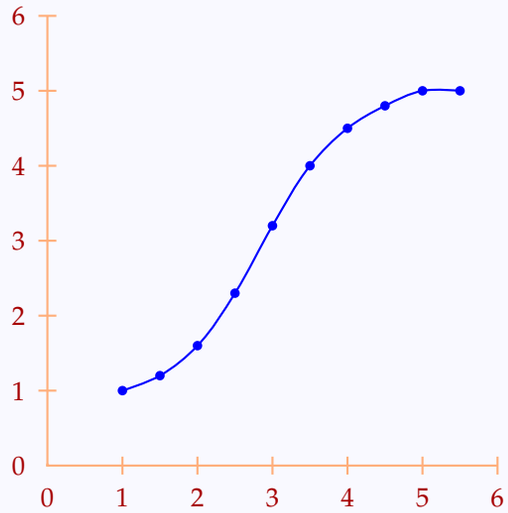
can also be used to produce the same picture.

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

The last command we describe for data plotting is `\listplot`. Here, we can give the data as coordinate pairs, *separated only by spaces*, as an argument to the command itself as in the example below:

```
\begin{pspicture}(0,0)(5,6)
      \renewcommand{%
        \pshlabel}[1]{%
          \color{Mahogany} #1}
      \renewcommand{%
        \psvlabel}[1]{%
          \color{Mahogany} #1}
        \psaxes[linecolor=Apricot]%
               (6,6)
        \listplot[plotstyle=curve,%
               showpoints=true,%
               linecolor=Blue]%
               {1 1
                1.5 1.2
                2 1.6
                2.5 2.3
                3 3.2
                3.5 4
                4 4.5
                4.5 4.8
                5 5
                5.5 5}
    \end{pspicture}
```



Here the data list is first expanded by TEX and then by PostScript. So, we can use PostScript commands to manipulate the data within this command. For example, we can draw the "inverse" graph of the above graph by interchanging the coordinates of each points using the exch command of PostScript.

Online LATEX Tutorial

Part II – Graphics

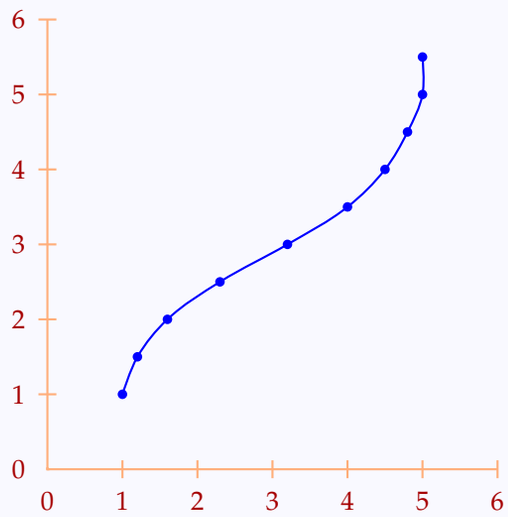
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA

http://www.tug.org.in

```
\begin{pspicture}(0,0)(5,6)
      \renewcommand{%
        \pshlabel}[1]{%
          \color{Mahogany} #1}
      \renewcommand{%
        \psvlabel}[1]{%
          \color{Mahogany} #1}
        \psaxes[linecolor=Apricot]%
               (6,6)
        \listplot[plotstyle=curve,%
               showpoints=true,%
               linecolor=Blue]%
               {1 1 exch
                1.5 1.2 exch
                2 1.6 exch
                2.5 2.3 exch
                3 3.2 exch
                3.5 4 exch
                4 4.5 exch
                4.5 4.8 exch
                5 5 exch
                5.5 5 exch}
    \end{pspicture}
```



In fact, the "list" given as an argument to `\listplot` may even be a PostScript program which generates a sequence of coordinate pairs. Again, as in the case of `\dataplot`, we can use the commands `\savedata` or `\readdata` with `\listplot` also, as illustrated below:

Online LaTeX Tutorial

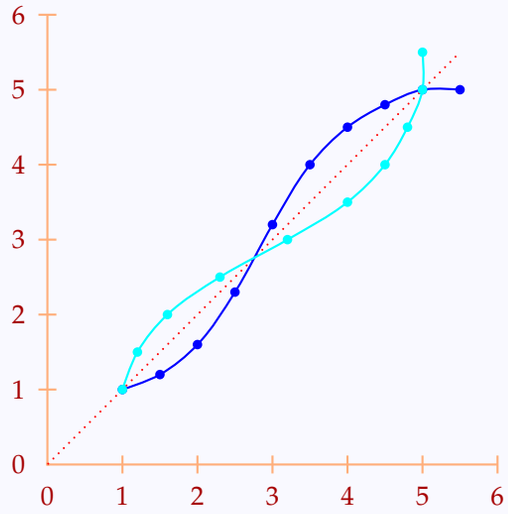
Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\savedata{\dirdata}[1 1 1.5 1.2 2 1.6 2.5 2.3
                    3 3.2 3.5 4 4 4.5 4.5 4.8
                    5 5 5.5 5]
\savedata{\invdata}[1 1 exch 1.5 1.2 exch
                    2 1.6 exch 2.5 2.3 exch
                    3 3.2 exch 3.5 4 exch
                    4 4.5 exch 4.5 4.8 exch
                    5 5 exch 5.5 5 exch]
\begin{pspicture}(0,0)(5,6.5)
  \renewcommand{%
    \pshlabel}[1]{%
    \color{Mahogany} #1}
  \renewcommand{%
    \psvlabel}[1]{%
    \color{Mahogany} #1}
  \psaxes[linecolor=Apricot]%
         (6,6)
  \psline[linestyle=dotted,%
          dotsep=2pt,%
          linecolor=Red]
         (0,0)(5.5,5.5)
  \listplot[plotstyle=curve,%
            showpoints=true,%
            linecolor=Blue]%
            {\dirdata}
  \listplot[plotstyle=curve,%
            showpoints=true,%
            linecolor=Cyan]%
            {\invdata}
 \end{pspicture}
```



Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in