pstricks - patch 15
new macros and bugfixes for pstricks

Herbert Voß∗ Rolf Niepraschk†

2004/05/12

Abstract

It is long time ago since pstricks.tex patch 14 came out. The new version patch 15 fixes some bugs and provides three new elliptic macros, which were already present in the old beta version of PSTricks.

There is also a new pstricks.sty, which makes the \pstrcol package obsolete. It uses the new color package xcolor, which provides a much more powerful color management than color.sty does. The pstricks.sty is a real \LaTeX{} package, it makes no sense for \TeX{} users. Nevertheless, using of pstrcol or package color is still possible.

Timothy Van Zandt was the one, who creates PSTricks, but Denis Girou was the one who makes it run over many years. Needless to say, how important his work is for PSTricks. Since more than nine month we are unable to get in touch with Denis, which is the reason why this update of PSTricks comes without any comments from Denis.

∗Herbert.Voss@perce.de
†Rolf.Niepraschk@ptb.de
1 New fill style eofill

PostScript has a special fillstyle, called eofill, which is now available with the option fillstyle=eofill. The following two images show the difference, the first one is filled with fillstyle=solid and the second one with the new option fillstyle=eofill.

2 Dashed lines

By default a dash line can be set with the option dash=<black> <white>, e.g. dash=10pt 5pt. This definition makes it impossible to define a dashed/dotted line. pstricks-add redefines this option for a use with four parameters dash=<black> <white> <black> <white>, where the last two can be omit. The following examples show different values for these parameters:
As seen in the above code, it is no problem to use dashed lines in the usual way with two parameters.

3 Ellipses

pstricks - patch 14 has only the following macro for drawing an ellipse:

\psellipse\([option]\)(x,y)(a,b)
\psellipse*[\[option\]](x,y)(a,b)

whith (x,y) as the center and (a,b) as the two radians (figure 1).
3.1 Ellipse based on pst-plot

With the \texttt{parametricplot} macro from \texttt{pst-plot} we can define a new macro for drawing ellipses:

\begin{verbatim}
% #1 options
% #2 a
% #3 b
% #4 start angle
% #5 end angle
\newcommand{\pstEllipse}[5][]{%
\psset{#1}
\parametricplot{#4}{#5}{#2 \texttt{space} t \texttt{cos mul} #3 \texttt{space} t \texttt{sin mul}}}
\end{verbatim}

which has the syntax

\texttt{\pstEllipse[<options>]{a}{b}{start angle}{end angle}}

This macro is not part of \texttt{pstricks.tex}, it is only defined for some demonstration.

\begin{verbatim}
\begin{pspicture}(-2.25,-1.75)
\psgrid
\pstEllipse[\
linewidth=0.2,\%
linecolor=red]{2}{1}{0}{360}
\pstEllipse[\
linewidth=0.1,arrows=\texttt{|->},\%
arrowsize=0.5,\%
linecolor=blue]{1}{2}{-30}{250}
\end{pspicture}
\end{verbatim}

Figure 2: The macro \texttt{\pstEllipse} which uses the \texttt{parametricplot} macro from \texttt{pst-plot}

As seen in figure 2 it is no problem to draw arcs of an ellipse. The center of these ellipses are by default \((0,0)\), with the \texttt{rput} macro it is also not a problem to put the ellipse anywhere in the coordinate system with any angle of rotating.
3.1.1 Wedge of an ellipse

To define a macro for a wedge of an ellipse (figure 3) is also easy with the \pscustom macro. which uses the following code:

\begin{verbatim}
\% #1 options
\% #2 a
\% #3 b
\% #4 start angle
\% #5 end angle
\newcommand\{\pstEllipseWedge\}[5][]\{
\psset{#1}
\pscustom{\%
\parametricplot{#4}{#5}{#2 space t cos mul #3 space t sin mul}\\
\psline(! #2 space #5 space cos mul #3 space #5 space sin mul)(0,0)\\
(! #2 space #4 space cos mul #3 space #4 space sin mul)\\
}
}\}
\end{verbatim}

This macro is also not part of of pstricks.tex, it is only defined for some demonstration.

\begin{verbatim}
\begin{pspicture}(-2.25,-1.75)
(2.25,1.75)
\psgrid
\pstEllipseWedge[linewidth=0.05,linestyle=red,fillstyle=hlines,fillcolor=red]{2}{1}{0}{130}
\pstEllipseWedge[linewidth=0.1,linestyle=blue,fillstyle=vlines,fillcolor=blue]{1}{2}{-30}{190}
\end{pspicture}
\end{verbatim}

Figure 3: The macro \pstEllipseWedge which uses the \parametricplot macro from pst-plot
3.2 New ellipse macros

All macros defined in this package are original from Timothy Van Zandt and Denis Girou and modified by several other authors. The available macros are

\psellipticarc[<options>]
\hspace{1cm}
\hspace{1cm}
\hspace{1cm}
\hspace{1cm}
\hspace{1cm}
\psellipticarcn[<options>]
\hspace{1cm}
\hspace{1cm}
\hspace{1cm}
\hspace{1cm}
\psellipticwedge[<options>]

3.2.1 Arc of an ellipse

Figure 4 shows different examples for this macro.

3.3 Arc of an ellipse with anti clockwise direction

Figure 5 shows different examples for this macro which is the same than the one figure ?? only drawn anti clockwise.

3.3.1 Wedge of an ellipse

Figure 6 shows different examples for this macro.

---

\begin{pspicture}(-2.25,-2.25)
\psellipticarc[linecolor=red](0,0)(0.5,1.5){30}{320}
\end{pspicture}

Figure 4: The macro \psellipticarc
4 \texttt{pstricks.sty}

In the past there were some problems with \texttt{pstricks.tex} and the package \texttt{color.sty}. \texttt{pstricks.sty} tried to get rid of them but not with success in any case. The new package \texttt{pstricks.sty} loads first \texttt{pstricks.tex}, does some modification to \texttt{pstricks}, loads \texttt{xcolor.sty} and some more modifications to the the code to get \texttt{pstricks} and colors work in a right way. It also renames the \texttt{\scalebox} macro to \texttt{\psscalebox} to prevent clashes with the one from the package \texttt{graphicx.sty} which has the same name but another syntax. If you want to use the macro from \texttt{graphicx}, then load this package as the last one.