

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 `\pstcol` 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 `pstcol` 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

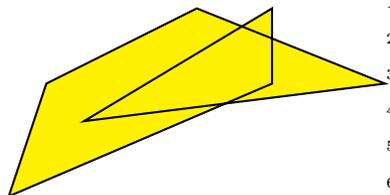
†Rolf.Niepraschk@ptb.de

Contents

1	New fill style eofill	3
2	Dashed lines	3
3	Ellipses	4
3.1	Ellipse based on <code>pst-plot</code>	5
3.1.1	Wedge of an ellipse	6
3.2	New ellipse macros	7
3.2.1	Arc of an ellipse	7
3.3	Arc of an ellipse with anti clockwise direction	7
3.3.1	Wedge of an ellipse	7
4	<code>pstricks.sty</code>	8

1 New fill style eofill

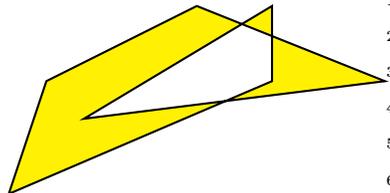
PostScript has a special fillstyle, called `eofill`, which is now available with the option `fillstyle=eofill`. The following two images show the difference, the first one is filled with `fillstyle=solid` and the second one with the new option `fillstyle=eofill`.



```

1 \begin{pspicture}(5,2.5)
2 \pspolygon[unit=0.5cm,%
3   fillstyle=solid,%
4   fillcolor=yellow](7,3)(0,0)(1,3)
5   (5,5)(10,3)(2,2)(7,5)(7,3)
6 \end{pspicture}

```



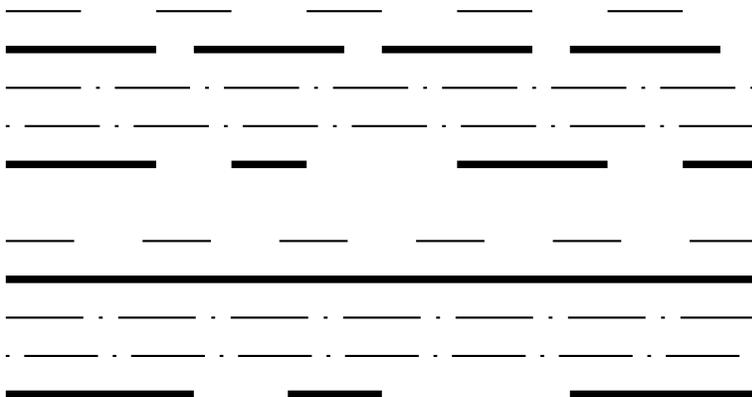
```

1 \begin{pspicture}(5,2.5)
2 \pspolygon[unit=0.5cm,%
3   fillstyle=eofill,%
4   fillcolor=yellow](7,3)(0,0)(1,3)%
5   (5,5)(10,3)(2,2)(7,5)(7,3)
6 \end{pspicture}

```

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:



```

1 {\psset{linestyle=dashed,dashadjust=false}
2 \psline[dash=1 1](0,0)(10,0)\
3 \psline[linewidth=1mm,dash=2 0.5](0,0)(10,0)\
4 \psline[dash=1 0.2 0.05 0.2](0,0)(10,0)\
5 \psline[dash=0.05 0.2 1 0.2](0,0)(10,0)\
6 \psline[linewidth=1mm,dash=2 1 1 2](0,0)(10,0)\
7
8 \psset{dashadjust=true}
9 \psline[dash=1 1](0,0)(10,0)\
10 \psline[linewidth=1mm,dash=2 0](0,0)(10,0)\
11 \psline[dash=1 0.2 0.05 0.2](0,0)(10,0)\
12 \psline[dash=0.05 0.2 1 0.2](0,0)(10,0)\
13 \psline[linewidth=1mm,dash=2 1 1 2](0,0)(10,0)}

```

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).

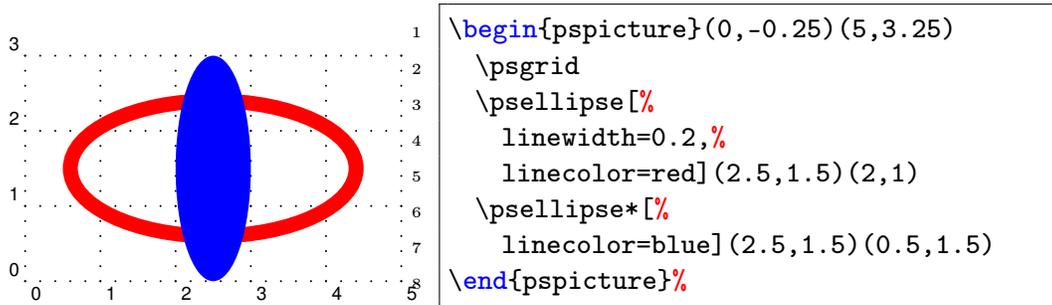


Figure 1: The `pstricks` macro `\psellipse`

3.1 Ellipse based on pst-plot

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

```
1 % #1 options
2 % #2 a
3 % #3 b
4 % #4 start angle
5 % #5 end angle
6 \newcommand{\pstEllipse}[5] []{%
7   \psset{#1}
8   \parametricplot{#4}{#5}{#2\space t cos mul #3\space t sin mul}}
```

which has the syntax

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

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

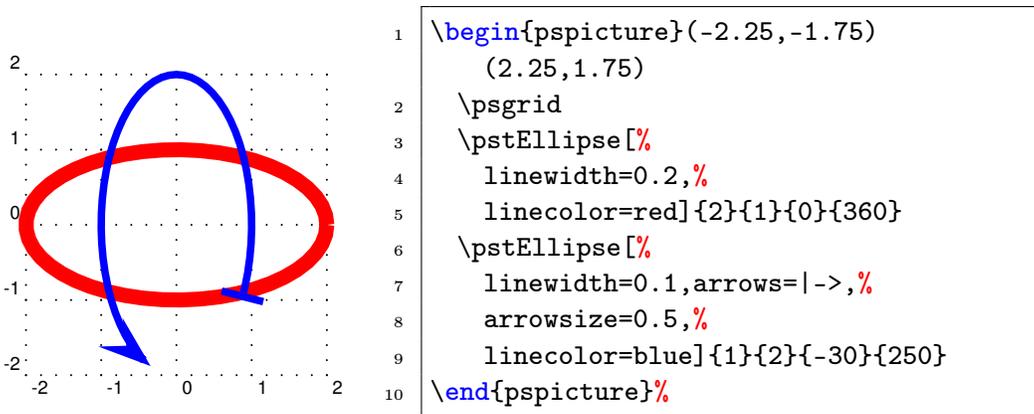


Figure 2: The macro `\pstEllipse` which uses the `\parametricplot` macro from `pst-plot`

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 `\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:

```

1 % #1 options
2 % #2 a
3 % #3 b
4 % #4 start angle
5 % #5 end angle
6 \newcommand{\pstEllipseWedge}[5] []{%
7   \psset{#1}
8   \pscustom{%
9     \parametricplot{#4}{#5}{#2\space t cos mul #3\space t sin mul}%
10    \psline(! #2\space #5\space cos mul #3\space #5\space sin mul)%
11    (0,0)%
12    (! #2\space #4\space cos mul #3\space #4\space sin mul)%
13    }%
14 }

```

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

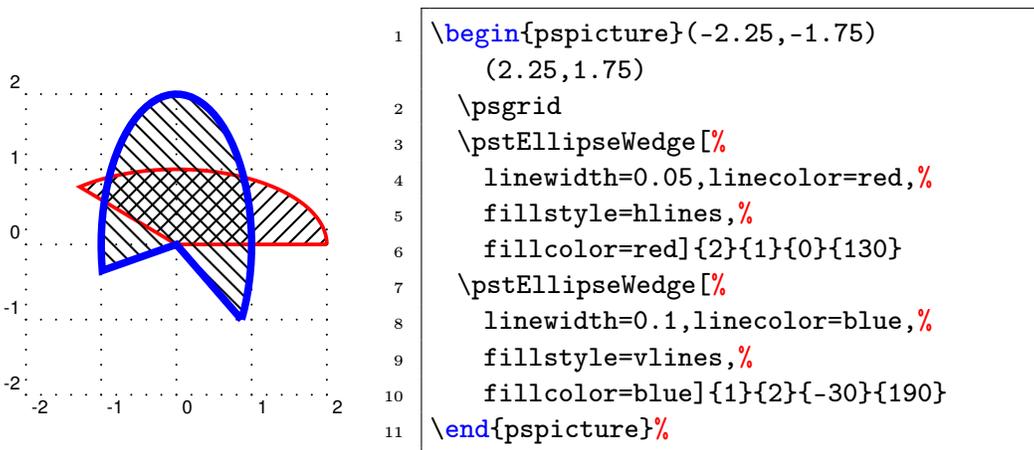


Figure 3: The macro `\pstEllipseWedge` which uses the `\parametricplot` macro from `pst-plot`

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>]
  {<arrows>}(<center>)(a,b){start angle}{end angle}
\psellipticarcn[<options>]
  {<arrows>}(<center>)(a,b){start angle}{end angle}
\psellipticwedge[<options>]
  {<arrows>}(<center>)(a,b){start angle}{end angle}
```

3.2.1 Arc of an ellipse

Figure 4 shows different examples for this macro.

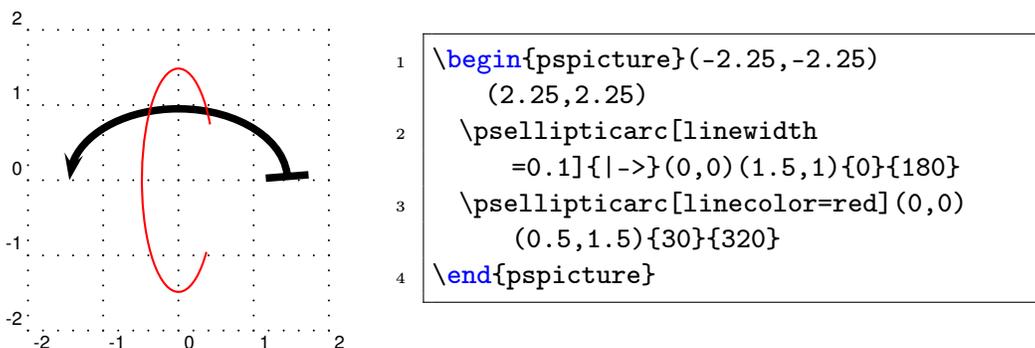


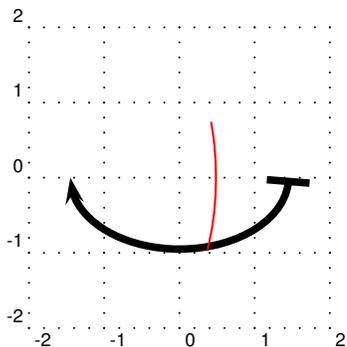
Figure 4: The macro `\psellipticarc`

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.

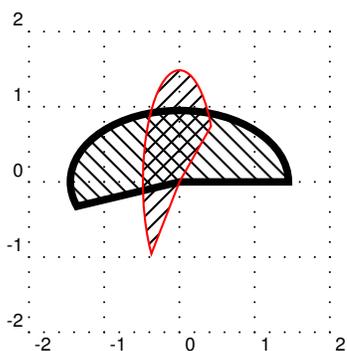


```

1 \begin{pspicture}(-2.25,-2.25)
   (2.25,2.25)
2 \psellipticarcn[linewidth
   =0.1]{|->}(0,0)(1.5,1){0}{180}
3 \psellipticarcn[linecolor=red](0,0)
   (0.5,1.5){30}{320}
4 \end{pspicture}

```

Figure 5: The macro `\psellipticarcn`



```

1 \begin{pspicture}(-2.25,-2.25)
   (2.25,2.25)
2 \psgrid
3 \psellipticwedge[%
4 fillstyle=vlines,%
5 linewidth=0.1](0,0)(1.5,1){0}{200}
6 \psellipticwedge[%
7 fillstyle=hlines,%
8 linecolor=red](0,0)(0.5,1.5){30}{220}
9 \end{pspicture}

```

Figure 6: The macro `\psellipticwedge`

4 pstricks.sty

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