PSTricks

pst2pdf

Running a PSTricks document with pdflatex;
v. 0.16

September 14, 2014

Package author(s):
Herbert Voß
Pablo González Luengo

Contents

1 Introduction

PSTricks as PostScript related package uses the programming language PostScript for internal calculations. This is an important advantage, because floating point arithmetic is no problem. Nearly all mathematical calculation can be done when running the DVI-file with Ghostscript. However, creating a PDF file in a direct way with pdflatex is not possible. pdflatex cannot understand the PostScript related stuff.

Instead of running pdflatex one can use the perl *script* pst2pdf, it extracts all PSTricks related code into single documents with the same preamble as the original main document.

The pst2pdf *script* runs document, clips all whitespace around the image and creates a .pdf images of the PSTricks related code.

In a last run which is the pdflatex the PSTricks code in the main document is replaced by the created images.

2 Requirements

2.1 Programs needed

pst2pdf needs a latest version Ghostscript (9.14), perl (>=5.18), pdf2svg, pdftoppm and pdftops (from poppler-utils or xpdf-utils) for the process file. If you need a create .pdf images whitout Ghostscript, use single mode (see 4.2).

2.2 Preparating file

The script scan the file for pspicture and postscript environments, which are then taken with its contents from the main file to create stand alone documents with the same preamble as the main document. The pspicture environment can be nested, the postscript one not! But it can contain an environment pspicture, but not vice versa. The postscript environment should always be used, when there is some code before a pspicture environment or for some code which is not inside of a pspicture environment.

pst2pdf delete al lines contains PSTricks package before last run, if you need delete other PSTricks code in preamble use:

```
%CleanPST
pstricks code
%CleanPST
```

3 Environments support

pst2pdf support fourth environments in default and single way:

```
\pspicture*          \begin{pspicture}    \begin{pspicture*}   \begin{postscript}
\psset{...}          \psset{...}          \psset{...}          \psset{...}
pstricks code        pstricks code        pstricks code        pstricks code
\endpspicture        \end{pspicture}      \end{pspicture*}     \end{postscript}
```

Note: When using the default mode, images are created using Ghostscript and preview package, in this case, it is not necessary to write psset into PSTricks environment.

4 Running the script

4.1 Default mode

The general syntax for the `perl` *script* is simple:

```
perl pst2pdf file.tex -options
```

For TEXLive users:

```
pst2pdf file.tex -options
```

In this way `pst2pdf` creates a new file called *file-pst.tex* and copy all `pspicture` and `postscript` environments, then processed and create file-pdf.pdf and file-fig-1.pdf, file-fig-2.pdf, file-fig-....pdf and file-fig-1.tex, file-fig-2.tex, file-fig-....tex for all `pspicture` and `postscript` using `Ghostscript`.

4.2 Single mode

If you do not have `Ghostscript` use the option `-np,--single` in this mode, the files are processed separately (take a more time to create images files). For example:

```
pst2pdf file.tex --single
```

create file-pdf.pdf and file-fig-1.pdf, file-fig-2.pdf, file-fig-....pdf and file-fig-1.tex, file-fig-2.tex, file-fig-....tex for all `pspicture` and `postscript` environments (see 5).

5 Options

The options listed in Table 1 refer only to the *script* and not the LATEX file.

For Help in command line use:

```
pst2pdf -help
```

6 Other image format

If you need to create other image formats use `pst2pdf`, move to images dir and use `mogrify` command (from `ImageMagick`), for examples:

```
mogrify -format tiff *.ppm
```

generate `.tiff` images files.

Table 1: Optional arguments for `pst2pdf`

name	*values*	*default*	*description*
`-h,--help`	boolean	`1`	print help and exit.
`-l,--license`	boolean	`0`	print license and exit.
`-v,--version`	boolean	`1`	print version and exit.
`-m,--margins`	literal	`1`	margins for pdfcrop (in bp).
`-d,--dpi`	integer	`300`	the dots per inch for a created `.ppm` file.
`-j,--jpg`	boolean	`0`	creates `.jpg` images (need `Ghostscript`).
`-p,--png`	boolean	`0`	creates `.png` images (need `Ghostscript`).
`-e,--eps`	boolean	`0`	creates `.eps` images (need `pdftops`).
`-s,--svg`	boolean	`0`	creates `.ppm` images (need `pdf2svg`).
`-P,--ppm`	boolean	`0`	creates `.ppm` images (need `pdftoppm`).
`-c,--clear`	boolean	`0`	delete all temporary files.
`-a,--all`	boolean	`0`	generte all images type and clear.
`-x,--xetex`	boolean	`0`	using `xelatex` instead of `latex` for process.
`-np,--single`	boolean	`0`	create images type (whitout Ghostscript).
`-ni,--noimages`	boolean	`0`	generate file-pdf.tex, but no images.
`-ns,--nosource`	boolean	`0`	delte all source for images.
`--imgdir`	literal	`images/`	the directory for the created images.
`--ignore`	literal	`other`	skip other verbatim environment.
`--Verbose`	boolean	`1`	for a long `pst2pdf` log.
`--bibtex`	boolean	`0`	runs `bibtex`.
`--bibtex`	boolean	`0`	runs `biber` if a file with extension `.bcf` exists.

References

[1] Denis Girou. Présentation de PSTricks. *Cahier GUTenberg*, 16:21–70, April 1994.

[2] Michel Goosens, Frank Mittelbach, Sebastian Rahtz, Denis Roegel, and Herbert Voß. *The LaTeX Graphics Companion*. Addison-Wesley Publishing Company, Reading, Mass., 2007.

[3] Laura E. Jackson and Herbert Voß. Die Plot-Funktionen von `pst-plot`. *Die TeXnische Komödie*, 2/02:27–34, June 2002.

[4] Nikolai G. Kollock. *PostScript richtig eingesetzt: vom Konzept zum praktischen Einsatz*. IWT, Vaterstetten, 1989.

[5] Herbert Voß. Die mathematischen Funktionen von PostScript. *Die TeXnische Komödie*, 1/02, March 2002.

[6] Herbert Voß. *PSTricks – Grafik für TeX und LaTeX*. DANTE – Lehmanns, Heidelberg/Hamburg, 6. edition, 2011.

[7] Timothy van Zandt. *`multido.tex` - a loop macro, that supports fixed-point addition*. CTAN:/graphics/pstricks/generic/multido.tex, 1997.

[8] Timothy van Zandt and Denis Girou. Inside PSTricks. *TUGboat*, 15:239–246, September 1994.

[9] Timothy van Zandt and Herbert Voß. *PSTricks - `PostScript` macros for generic TeX*. http://PSTricks.tug.org/, 2011.

[10] Timothy van Zandt and Herbert Voß. *`pst-plot`: Plotting two dimensional functions and data*. CTAN:graphics/pstricks/generic/pst-plot.tex, 2011.

Index