The **asymptote** package

John Bowman, Tom Prince, and Will Robertson

2016/11/26   v1.33

Abstract

This package provides integration of inline and external Asymptote graphics within a LaTeX document.

Contents

1   Introduction

This is the documentation for the LaTeX package `asymptote` which accompanies the Asymptote drawing package. For further details on Asymptote, please see its documentation in `asymptote.pdf`.

2   User syntax

2.1   Package loading and options

The package may take two options at load time: `inline` or `attach`. These options can also be set at any time with the `\asysetup{⟨options⟩}` command, or specified individually in the optional argument to each `asy` environment or `asyinclude` command.

The `inline` option uses Asymptote's 'inline' mode whereby included graphics have their labels typeset in the environment of the document they are contained within. Otherwise the Asymptote graphics are self-contained and their formatting is independent of the document.

The `attach` option allows generated graphics to be embedded within the PDF using the `attachfile2` package; please load that package separately if you wish to use it. The `attach` option takes precedence over the `inline` option.

This package produces quite a number of output files, which by default are created in the same directory as the LaTeX document that is being compiled. To keep things more tidy, you can specify an output directory for these files by defining the `\asydir` command. For example, if you wish to store the figure files in the subdirectory `asytmp/`, the you would write `\renewcommand\asydir{asytmp}`.

Alternatively (and tentatively), you may write `dir=asytmp` in either the `asy` environment options or the options to `\asysetup`.

2.2 Commands for inserting Asymptote graphics

The main environment defined by the package is the `asy` environment, in which verbatim Asymptote code is placed that will be compiled for generating a graphic in the document. For example,

```
\begin{figure}
\begin{asy}[ <options> ]
<ASYMPTOTE CODE>
\end{asy}
\caption{...}\label{...}
```

If you have Asymptote code in a separate file, you can include it with the `\asyinclude[⟨options⟩]{⟨filename⟩}` command.

For Asymptote code that should be included in *every* graphic, define it using the `asydef` environment.

2.3 Graphics options

Both the `asy` environment and the `\asyinclude` command take optional parameters for controlling aspects of the graphics creation. In addition to locally setting `inline` and `attach`, the following options may also be used:

width Width of the figure

height Height of the figure

keepAspect Maintain aspect ratio [default true]

viewportwidth Viewport width for 3D figures

viewportheight Viewport height for 3D figures

These may also be set globally using the `\asysetup` command.

3 Processing the document

After running LaTeX on the document, it is necessary to process the Asymptote graphics so they can be included in the next compilation. The simplest procedure is a recipe such as

```
pdflatex mydoc
asy mydoc-*.asy
pdflatex mydoc
```

This technique will recompile each graphic every time, however. To only recompile graphics that have changed, use the `latexmk` tool. Asymptote is distributed with a `latexmkrc` configuration file; place this file in a place where `latexmk` will find it and your document may be compiled, including the `asy` compilations, with `latexmk mydoc` or `latexmk --pdf mydoc`.

4   Implementation

```
1 \def\Asymptote{{\tt Asymptote}}
```

```
2 \InputIfFileExists{\jobname.pre}{}{}
```

4.1   Allocations

Allocations

```
3 \newbox\ASYbox
4 \newdimen\ASYdimen
5 \newcounter{asy}
```

```
6 \newwrite\AsyStream
7 \newwrite\AsyPreStream
```

```
8  \newif\ifASYinline
9  \newif\ifASYattach
10 \newif\ifASYkeepAspect
11 \ASYkeepAspecttrue
```

4.2   Packages

```
12 \RequirePackage{keyval}
13 \RequirePackage{ifthen}
14 \RequirePackage{color,graphicx}
```

Emulating packages   We cannot assume that Asymptote users have recent TEX distributions. (E.g., Fedora until recently still shipped teTeX.) So load ifpdf and ifxetex if they exist; otherwise, emulate them.

In due course, delete this code and just load the packages.

```
15 \IfFileExists{ifpdf.sty}{
16   \RequirePackage{ifpdf}
17 }{
18   \expandafter\newif\csname ifpdf\endcsname
19   \ifx\pdfoutput\@undefined\else
20     \ifcase\pdfoutput\else
21       \pdftrue
22     \fi
23   \fi
24 }
```

```
25 \IfFileExists{ifxetex.sty}{
26   \RequirePackage{ifxetex}
27 }{
28   \expandafter\newif\csname ifxetex\endcsname
29   \ifx\XeTeXversion\@undefined\else
30     \xetextrue
31   \fi
32 }
```

\CatchFileDef Used for \asyinclude. Note that the fallback definition is not as robust as the one provided by catchfile.

```
33 \IfFileExists{catchfile.sty}{
34   \RequirePackage{catchfile}
35 }{
36   \newcommand\CatchFileDef[3]{%
37     \begingroup
38     \everyeof{%
39       \ENDCATCHFILEMARKER
40       \noexpand
41     }%
42     \long\def\@tempa####1\ENDCATCHFILEMARKER{%
43       \endgroup
44       \def##1{####1}%
45     }%
46     ##3%
47     \expandafter\@tempa\@@input ##2\relax
48   }
49 }
```

Ensuring attachfile2 is loaded if [attach] is requested

```
50 \newif\if@asy@attachfile@loaded

51 \AtBeginDocument{%
52   \@ifpackageloaded{attachfile2}{\@asy@attachfile@loadedtrue}{}%
53   \let\asy@check@attachfile\asy@check@attachfile@loaded
54 }

55 \newcommand\asy@check@attachfile@loaded{%
56   \if@asy@attachfile@loaded\else
57     \PackageError{asymptote}{You must load the attachfile2 package}{^^J%
58       You have requested the [attach] option for some or all of your^^J%
59       Asymptote graphics, which requires the attachfile2 package.^^J%
60       Please load it in the document preamble.^^J%
61     }%
62   \fi
63 }

64 \newcommand\asy@check@attachfile{%
65   \AtBeginDocument{\asy@check@attachfile@loaded}%
66   \let\asy@check@attachfile\@empty
67 }
```

Macros

```
68 \def\csarg#1#2{\expandafter#1\csname#2\endcsname}
```

4.3   Package options

```
69 \DeclareOption{inline}{%
70   \ASYinlinetrue
```

```
71 }
72 \DeclareOption{attach}{%
73   \asy@check@attachfile
74   \ASYattachtrue
75 }
76 \ProcessOptions*

77 \def\asylatexdir{}
78 \def\asydir{}
79 \def\ASYasydir{}
80 \def\ASYprefix{}
```

4.4   Testing for PDF output

Note this is not quite the same as `\ifpdf`, since we still want PDF output when using XeTeX.

```
81 \newif\ifASYPDF
82 \ifxetex
83   \ASYPDFtrue
84   \usepackage{everypage}
85 \else
86   \ifpdf
87     \ASYPDFtrue
88   \fi
89 \fi
90 \ifASYPDF
91   \def\AsyExtension{pdf}
92 \else
93   \def\AsyExtension{eps}
94 \fi
```

4.5   Bug squashing

```
95 \def\unquoteJobname#1"#2"#3\relax{%
96   \def\rawJobname{#1}%
97   \ifx\rawJobname\empty
98     \def\rawJobname{#2}%
99   \fi
100 }
101 \expandafter\unquoteJobname\jobname""\relax
```

Work around jobname bug in MiKTeX 2.5 and 2.6: Turn stars in file names (resulting from spaces, etc.) into minus signs

```
102 \def\fixstar#1*#2\relax{%
103   \def\argtwo{#2}%
104   \ifx\argtwo\empty
105     \gdef\Jobname{#1}%
106   \else
107     \fixstar#1-#2\relax
108   \fi
109 }
```

```
110 \expandafter\fixstar\rawJobname*\relax
```

Work around bug in dvips.def: allow spaces in file names.

```
111 \def\Ginclude@eps#1{%
112   \message{<#1>}%
113   \bgroup
114   \def\@tempa{!}%
115   \dimen@\Gin@req@width
116   \dimen@ii.1bp\relax
117   \divide\dimen@\dimen@ii
118   \@tempdima\Gin@req@height
119   \divide\@tempdima\dimen@ii
120     \special{PSfile=#1\space
121       llx=\Gin@llx\space
122       lly=\Gin@lly\space
123       urx=\Gin@urx\space
124       ury=\Gin@ury\space
125       \ifx\Gin@scalex\@tempa\else rwi=\number\dimen@\space\fi
126       \ifx\Gin@scaley\@tempa\else rhi=\number\@tempdima\space\fi
127       \ifGin@clip clip\fi}%
128   \egroup
129 }
```

4.6   Input/Output

```
130 \immediate\openout\AsyPreStream=\jobname.pre\relax
131 \AtEndDocument{\immediate\closeout\AsyPreStream}

132 \def\WriteAsyLine#1{%
133   \immediate\write\AsyStream{\detokenize{#1}}%
134 }

135 \def\globalASYdefs{}
136 \def\WriteGlobalAsyLine#1{%
137   \expandafter\g@addto@macro
138   \expandafter\globalASYdefs
139   \expandafter{\detokenize{#1^^J}}%
140 }
```

4.7   Commands for verbatim processing environments

```
141 \def\ProcessAsymptote#1{%
142   \begingroup
143   \def\CurrentAsymptote{#1}%
144   \let\do\@makeother \dospecials
145   \@makeother\^^L% and whatever other special cases
146   \catcode`\ =10
147   \endlinechar`\^^M \catcode`\^^M=12 \xAsymptote
148 }
```

Need lots of comment chars here because ⟨*line end*⟩ is no longer a space character.

```
149 \begingroup
150   \catcode`\^^M=12 \endlinechar=-1\relax%
151   \gdef\xAsymptote{%
```

```
152     \expandafter\ProcessAsymptoteLine%
153   }
154   \gdef\ProcessAsymptoteLine#1^^M{%
155     \def\@tempa{#1}%
156     {%
157       \escapechar=-1\relax%
158       \xdef\@tempb{\string\\end\string\{\CurrentAsymptote\string\}}%
159     }%
160     \ifx\@tempa\@tempb%
161       \edef\next{\endgroup\noexpand\end{\CurrentAsymptote}}%
162     \else%
163       \ThisAsymptote{#1}%
164       \let\next\ProcessAsymptoteLine%
165     \fi%
166     \next%
167   }
168 \endgroup
169 \def\asy@init{%
170   \def\ASYlatexdir{}%
171   \ifx\asylatexdir\empty\else
172     \def\ASYlatexdir{\asylatexdir/}%
173   \fi
174   \ifx\asydir\empty\else
175     \def\ASYasydir{\asydir/}%
176   \fi
177   \def\ASYprefix{\ASYlatexdir\ASYasydir}%
178 }
```

4.8   User interface

```
179 \newcommand\asy[1][]{%
180   \stepcounter{asy}%
181   \setkeys{ASYkeys}{#1}%
```

Disable the "inline" option if "attach" is enabled:
```
182   \ifASYattach
183     \ASYinlinefalse
184   \fi
185   \asy@init
186   \immediate\write\AsyPreStream{%
187     \noexpand\InputIfFileExists{%
188       \ASYprefix\noexpand\jobname-\the\c@asy.pre}{}{}%
189   }%
190   \asy@write@graphic@header
191   \let\ThisAsymptote\WriteAsyLine
192   \ProcessAsymptote{asy}%
193 }
194 \def\endasy{%
195   \asy@finalise@stream
196   \asy@input@graphic
197 }
```

```
198 \def\asy@write@graphic@header{%
199   \immediate\openout\AsyStream=\ASYasydir\jobname-\the\c@asy.asy\relax
200   \gdef\AsyFile{\ASYprefix\Jobname-\the\c@asy}%
201   \immediate\write\AsyStream{%
202     if(!settings.multipleView) settings.batchView=false;^^J%
203     \ifxetex
204       settings.tex="xelatex";^^J%
205     \else\ifASYPDF
206         settings.tex="pdflatex";^^J%
207     \fi\fi
208     \ifASYinline
209       settings.inlinetex=true;^^J%
210       deletepreamble();^^J%
211     \fi
212     defaultfilename="\Jobname-\the\c@asy";^^J%
213     if(settings.render < 0) settings.render=4;^^J%
214     settings.outformat="";^^J%
215     \ifASYattach
216       settings.inlineimage=false;^^J%
217       settings.embed=false;^^J%
218       settings.toolbar=true;^^J%
219     \else
220       settings.inlineimage=true;^^J%
221       settings.embed=true;^^J%
222       settings.toolbar=false;^^J%
223       viewportmargin=(2,2);^^J%
224     \fi
225     \globalASYdefs
226   }%
227 }
228 \def\asy@expand@keepAspect{%
229   \ifASYkeepAspect keepAspect=true%
230   \else keepAspect=false%
231   \fi%
232 }

233 \def\asy@finalise@stream{%
```

Setting `size()`. Only inserted if one of the dimensions is set explicitly (i.e., if both height and width are not empty).

```
234   \ifx\ASYwidth\@empty
235     \ifx\ASYheight\@empty
236       % write nothing!
237     \else
238       \immediate\write\AsyStream{size(0,\ASYheight,\asy@expand@keepAspect);}%
239     \fi
240   \else
241     \ifx\ASYheight\@empty
242       \immediate\write\AsyStream{size(\ASYwidth,0,\asy@expand@keepAspect);}%
243     \else
244       \immediate\write\AsyStream{size(\ASYwidth,\ASYheight,\asy@expand@keepAspect);}%
```

```
245     \fi
246   \fi
```

Setting `viewportsize=()`. Same logic as for `size()`.

```
247   \ifx\ASYviewportwidth\@empty
248     \ifx\ASYviewportheight\@empty
249       % write nothing!
250     \else
251       \immediate\write\AsyStream{viewportsize=(0,\ASYviewportheight);}%
252     \fi
253   \else
254     \ifx\ASYviewportheight\@empty
255       \immediate\write\AsyStream{viewportsize=(\ASYviewportwidth,0);}%
256     \else
257       \immediate\write\AsyStream{%
258         viewportsize=(\ASYviewportwidth,\ASYviewportheight);}%
259     \fi
260   \fi
261   \immediate\closeout\AsyStream
262 }
263 \def\asy@input@graphic{%
264   \ifASYinline
265     \IfFileExists{"\AsyFile.tex"}{%
266       \catcode`:=12\relax
267       \@@input"\AsyFile.tex"\relax
268     }{%
269       \PackageWarning{asymptote}{file `\AsyFile.tex' not found}%
270     }%
271   \else
272     \IfFileExists{"\AsyFile.\AsyExtension"}{%
273       \ifASYattach
274         \ifASYPDF
275           \IfFileExists{"\AsyFile+0.pdf"}{%
276             \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile+0".pdf}}%
277           }{%
278             \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile".pdf}}%
279           }%
280         \else
281           \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile.eps"}}%
282         \fi
283         \textattachfile{\AsyFile.\AsyExtension}{\phantom{\copy\ASYbox}}%
284         \vskip-\ht\ASYbox
285         \indent
286         \box\ASYbox
287       \else
288         \ifASYPDF
289           \includegraphics[hiresbb]{"\AsyFile".pdf}%
290         \else
291           \includegraphics[hiresbb]{"\AsyFile.eps"}%
292         \fi
```

9

```
293       \fi
294    }{%
```
3D PRC figures require inline mode.
```
295       \IfFileExists{"\AsyFile.tex"}{%
296          \catcode`:=12
297          \@@input"\AsyFile.tex"\relax
298       }{%
299          \PackageWarning{asymptote}{%
300             file `\AsyFile.\AsyExtension' not found%
301          }%
302       }%
303    }%
304    \fi
305 }
306 \def\asydef{%
307    \let\ThisAsymptote\WriteGlobalAsyLine
308    \ProcessAsymptote{asydef}%
309 }
310 \newcommand\asyinclude[2][]{%
311    \begingroup
312    \stepcounter{asy}%
313    \setkeys{ASYkeys}{#1}%
314    \ifASYattach
315       \ASYinlinefalse
316    \fi
317    \asy@init
318    \immediate\write\AsyPreStream{%
319       \noexpand\InputIfFileExists{%
320          \ASYprefix\noexpand\jobname-\the\c@asy.pre}{}{}%
321    }%
322    \asy@write@graphic@header
323    \IfFileExists{#2.asy}{%
324       \CatchFileDef\@tempa{#2.asy}{%
325          \let\do\@makeother
326          \dospecials
327          \endlinechar=10\relax
328       }%
329    }{%
330       \IfFileExists{#2}{%
331          \CatchFileDef\@tempa{#2}{%
332             \let\do\@makeother
333             \dospecials
334             \endlinechar=10\relax
335          }%
336       }{%
337          \PackageWarning{asymptote}{file #2 not found}%
338          \def\@tempa{}%
339       }%
340    }%
```

```
341    \immediate\write\AsyStream{\unexpanded\expandafter{\@tempa}}%
342    \asy@finalise@stream
343    \asy@input@graphic
344    \endgroup
345 }

346 \newcommand{\ASYanimategraphics}[5][]{%
347    \IfFileExists{_#3.pdf}{%
348       \animategraphics[{#1}]{#2}{_#3}{#4}{#5}%
349    }{}%
350 }
```

4.9   Keys for graphics processing

```
351 \newcommand\asysetup[1]{\setkeys{ASYkeys}{#1}}

352 \define@key{ASYkeys}{dir}{%
353    \def\asydir{#1}%
354 }
355 \def\ASYwidth{}
356 \define@key{ASYkeys}{width}{%
357    \edef\ASYwidth{\the\dimexpr#1\relax}%
358 }
359 \def\ASYheight{}
360 \define@key{ASYkeys}{height}{%
361    \edef\ASYheight{\the\dimexpr#1\relax}%
362 }
363 \define@key{ASYkeys}{keepAspect}[true]{%
364    \ifthenelse{\equal{#1}{true}}
365       {\ASYkeepAspecttrue}
366       {\ASYkeepAspectfalse}%
367 }
368 \def\ASYviewportwidth{}
369 \define@key{ASYkeys}{viewportwidth}{%
370    \edef\ASYviewportwidth{\the\dimexpr#1\relax}%
371 }
372 \def\ASYviewportheight{}
373 \define@key{ASYkeys}{viewportheight}{%
374    \edef\ASYviewportheight{\the\dimexpr#1\relax}%
375 }

376 \define@key{ASYkeys}{inline}[true]{%
377    \ifthenelse{\equal{#1}{true}}
378       {\ASYinlinetrue}
379       {\ASYinlinefalse}%
380 }
381 \define@key{ASYkeys}{attach}[true]{%
382    \ifthenelse{\equal{#1}{true}}
383       {\ASYattachtrue}
384       {\ASYattachfalse}%
385 }
```