

The `asymptote` package

John Bowman, Tom Prince, and Will Robertson

2019/11/04 v1.34

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/`, then 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 `\asyssetup` 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}
15 \RequirePackage[space]{grffile}
```

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.

```
16 \IfFileExists{ifpdf.sty}{
17   \RequirePackage{ifpdf}
18 }{
19   \expandafter\newif\csname ifpdf\endcsname
20   \ifx\pdfoutput\@undefined\else
21     \ifcase\pdfoutput\else
22       \pdftrue
23     \fi
24   \fi
25 }

26 \IfFileExists{ifxetex.sty}{
27   \RequirePackage{ifxetex}
28 }{
29   \expandafter\newif\csname ifxetex\endcsname
30   \ifx\XeTeXversion\@undefined\else
31     \xetextrue
32   \fi
33 }
```

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

```

34 \IfFileExists{catchfile.sty}{
35   \RequirePackage{catchfile}
36 }{
37   \newcommand\CatchFileDef[3]{%
38     \begingroup
39     \everyeof{%
40       \ENDCATCHFILEMARKER
41       \noexpand
42     }%
43     \long\def\@tempa####1\ENDCATCHFILEMARKER{%
44       \endgroup
45       \def##1{####1}%
46     }%
47     ##3%
48     \expandafter\@tempa\@input ##2\relax
49   }
50 }

```

Ensuring `attachfile2` is loaded if `[attach]` is requested

```

51 \newif\if@asy@attachfile@loaded

52 \AtBeginDocument{%
53   \@ifpackageloaded{attachfile2}{\@asy@attachfile@loadedtrue}{}%
54   \let\asy@check@attachfile\asy@check@attachfile@loaded
55 }

56 \newcommand\asy@check@attachfile@loaded{%
57   \if@asy@attachfile@loaded\else
58     \PackageError{asyptote}{You must load the attachfile2 package}{^^J%
59       You have requested the [attach] option for some or all of your^^J%
60       Asymptote graphics, which requires the attachfile2 package.^^J%
61       Please load it in the document preamble.^^J%
62     }%
63   \fi
64 }

65 \newcommand\asy@check@attachfile{%
66   \AtBeginDocument{\asy@check@attachfile@loaded}%
67   \let\asy@check@attachfile\@empty
68 }

```

Macros

```

69 \def\csarg#1#2{\expandafter#1\csname#2\endcsname}

```

4.3 Package options

```

70 \DeclareOption{inline}{%
71   \ASYinlinetrue

```

```

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

78 \def\asylatexdir{}
79 \def\asydir{}
80 \def\ASYasydir{}
81 \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.

```

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

```

4.5 Bug squashing

```

96 \def\unquoteJobname#1"#2"#3\relax{%
97   \def\rawJobname{#1}%
98   \ifx\rawJobname\empty
99     \def\rawJobname{#2}%
100   \fi
101 }
102 \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

```

103 \def\fixstar#1*#2\relax{%
104   \def\argtwo{#2}%
105   \ifx\argtwo\empty
106     \gdef\Jobname{#1}%
107   \else
108     \fixstar#1-#2\relax
109   \fi
110 }

```

```

111 \expandafter\fixstar\rawJobname*\relax
    Work around bug in grffile.  Work around bug in dvips.def: allow spaces in
    file names.
112 \def\Ginclude@eps#1{%
113   \message{<#1>}%
114   \bgroup
115   \def\@tempa{!}%
116   \dimen@\Gin@req@width
117   \dimen@ii.1bp\relax
118   \divide\dimen@\dimen@ii
119   \@tempdima\Gin@req@height
120   \divide\@tempdima\dimen@ii
121   \special{PSfile=#1\space
122     llx=\Gin@llx\space
123     lly=\Gin@lly\space
124     urx=\Gin@urx\space
125     ury=\Gin@ury\space
126     \ifx\Gin@scalex\@tempa\else rwi=\number\dimen@\space\fi
127     \ifx\Gin@scaley\@tempa\else rhi=\number\@tempdima\space\fi
128     \ifGin@clip clip\fi}%
129   \egroup
130 }

```

4.6 Input/Output

```

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

133 \def\WriteAsyLine#1{%
134   \immediate\write\AsyStream{\detokenize{#1}}%
135 }

136 \def\globalASYdefs{}
137 \def\WriteGlobalAsyLine#1{%
138   \expandafter\g@addto@macro
139   \expandafter\globalASYdefs
140   \expandafter{\detokenize{#1^^J}}%
141 }

```

4.7 Commands for verbatim processing environments

```

142 \def\ProcessAsymptote#1{%
143   \begingroup
144   \def\CurrentAsymptote{#1}%
145   \let\do\@makeother \dospecials
146   \@makeother^^L% and whatever other special cases
147   \catcode'\ =10
148   \endlinechar'\^^M \catcode'\^^M=12 \xAsymptote
149 }

```

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

```

150 \begingroup
151   \catcode'\^^M=12 \endlinechar=-1\relax%

```

```

152 \gdef\xAsymptote{%
153   \expandafter\ProcessAsymptoteLine%
154 }
155 \gdef\ProcessAsymptoteLine#1^^M{%
156   \def\@tempa{#1}%
157   {%
158     \escapechar=-1\relax%
159     \xdef\@tempb{\string\end\string\{\CurrentAsymptote\string\}}%
160   }%
161   \ifx\@tempa\@tempb%
162     \edef\next{\endgroup\noexpand\end{\CurrentAsymptote}}%
163   \else%
164     \ThisAsymptote{#1}%
165     \let\next\ProcessAsymptoteLine%
166   \fi%
167   \next%
168 }
169 \endgroup
170 \def\asy@init{%
171   \def\ASYlatexdir{}%
172   \ifx\asylatexdir\empty\else
173     \def\ASYlatexdir{\asylatexdir/}%
174   \fi
175   \ifx\asydir\empty\else
176     \def\ASYasydir{\asydir/}%
177   \fi
178   \def\ASYprefix{\ASYlatexdir\ASYasydir}%
179 }

```

4.8 User interface

```

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

```

Disable the "inline" option if "attach" is enabled:

```

183   \ifASYattach
184     \ASYinlinefalse
185   \fi
186   \asy@init
187   \immediate\write\AsyPreStream{%
188     \noexpand\InputIfFileExists{%
189       \ASYprefix\noexpand\jobname-\the\c@asy.pre}{-}{-}%
190   }%
191   \asy@write@graphic@header
192   \let\ThisAsymptote\WriteAsyLine
193   \ProcessAsymptote{asy}%
194 }

```

```

195 \def\endasy{%
196   \asy@finalise@stream
197   \asy@input@graphic

```

```

198 }

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

229 \def\asy@expand@keepAspect{%
230   \ifASYkeepAspect keepAspect=true%
231   \else keepAspect=false%
232   \fi%
233 }

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

```



```

244     \else
245         \immediate\write\AsyStream{size(\ASYwidth,\ASYheight,\asy@expand@keepAspect);}%
246     \fi
247 \fi

Setting viewportsize=(). Same logic as for size().
248 \ifx\ASYviewportwidth\@empty
249     \ifx\ASYviewportheight\@empty
250         % write nothing!
251     \else
252         \immediate\write\AsyStream{viewportsize=(0,\ASYviewportheight);}%
253     \fi
254 \else
255     \ifx\ASYviewportheight\@empty
256         \immediate\write\AsyStream{viewportsize=(\ASYviewportwidth,0);}%
257     \else
258         \immediate\write\AsyStream{%
259             viewportsize=(\ASYviewportwidth,\ASYviewportheight);}%
260     \fi
261 \fi
262 \immediate\closeout\AsyStream
263 }

264 \def\asy@input@graphic{%
265     \ifASYinline
266         \IfFileExists{"\AsyFile.tex"}{%
267             \catcode'\:=12\relax
268             \@@input"\AsyFile.tex"\relax
269         }{%
270             \PackageWarning{asymptote}{file '\AsyFile.tex' not found}%
271         }%
272     \else
273         \IfFileExists{"\AsyFile.\AsyExtension"}{%
274             \ifASYattach
275                 \ifASYPDF
276                     \IfFileExists{"\AsyFile+0.pdf"}{%
277                         \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{\AsyFile+0.pdf}}%
278                     }{%
279                         \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{\AsyFile.pdf}}%
280                     }%
281                 \else
282                     \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{\AsyFile.eps}}%
283                 \fi
284                 \textattachfile{\AsyFile.\AsyExtension}{\phantom{\copy\ASYbox}}%
285                 \vskip-\ht\ASYbox
286                 \indent
287                 \box\ASYbox
288             \else
289                 \ifASYPDF
290                     \includegraphics[hiresbb]{\AsyFile.pdf}%
291                 \else

```

```

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

```

```

340 }%
341 }%
342 \immediate\write\AsyStream{\unexpanded\expandafter{\@tempa}}%
343 \asy@finalise@stream
344 \asy@input@graphic
345 \endgroup
346 }

347 \newcommand{\ASYanimategraphics}[5][{}]{%
348   \IfFileExists{_#3.pdf}{%
349     \animategraphics[{#1}]{#2}{_#3}{#4}{#5}%
350   }{}%
351 }

```

4.9 Keys for graphics processing

```

352 \newcommand\asysetup[1]{\setkeys{ASYkeys}{#1}}

353 \define@key{ASYkeys}{dir}{%
354   \def\asydir{#1}%
355 }
356 \def\ASYwidth{}
357 \define@key{ASYkeys}{width}{%
358   \edef\ASYwidth{\the\dimexpr#1\relax}%
359 }
360 \def\ASYheight{}
361 \define@key{ASYkeys}{height}{%
362   \edef\ASYheight{\the\dimexpr#1\relax}%
363 }
364 \define@key{ASYkeys}{keepAspect}[true]{%
365   \ifthenelse{\equal{#1}{true}}
366     {\ASYkeepAspecttrue}
367     {\ASYkeepAspectfalse}%
368 }
369 \def\ASYviewportwidth{}
370 \define@key{ASYkeys}{viewportwidth}{%
371   \edef\ASYviewportwidth{\the\dimexpr#1\relax}%
372 }
373 \def\ASYviewportheight{}
374 \define@key{ASYkeys}{viewportheight}{%
375   \edef\ASYviewportheight{\the\dimexpr#1\relax}%
376 }
377 \define@key{ASYkeys}{inline}[true]{%
378   \ifthenelse{\equal{#1}{true}}
379     {\ASYinlinetrue}
380     {\ASYinlinefalse}%
381 }
382 \define@key{ASYkeys}{attach}[true]{%
383   \ifthenelse{\equal{#1}{true}}
384     {\ASYattachtrue}
385     {\ASYattachfalse}%

```

386 }