

The `apxproof` package

Pierre Senellart
`pierre@senellart.com`

2016/10/27 v1.0.0-dev

Abstract

This package facilitates the writing of scientific article with proofs deferred to the appendix.

1 Usage

TODO

2 Implementation

We now describe the entire code of the package, in literate programming fashion. Throughout the package, we use the `axp@` prefix to identify local macros and environment names, not meant to be used by the final user.

2.1 Dependencies

We first load a few package dependencies:

- `bibunits` to add a second bibliography for the appendix material.

```
1 \RequirePackage{bibunits}
```

- `environ` to easily define the repeated theorem environments.

```
2 \RequirePackage{environ}
```

- `etoolbox` to define simple toggles.

```
3 \RequirePackage{etoolbox}
```

- `fancyvrb` for the bulk of the work of exporting appendix material in an auxiliary file.

```
4 \RequirePackage{fancyvrb}
```

- `ifthen` for easier comparison of character strings.

```
5 \RequirePackage{ifthen}
```

- `kvoptions` to manage options passed to the package.

```
6 \RequirePackage{kvoptions}
```

- `amsthm` for its `\newtheorem` macro. Some document classes (e.g., `lipics`) preload `amsthm`: this is fine, `\RequirePackage{amsthm}` will simply have no effect. On the other hand, some other document classes (e.g., `llncs` or `sig-alternate`) define a `proof` environment that conflicts with `amsthm`, so we have to undefine this environment before loading `amsthm`.

```
7 \@ifpackageloaded{amsthm}{
8   }{
9     \let\proof\undefined
10    \let\endproof\undefined
11  }
12 \RequirePackage{amsthm}
```

2.2 Option Processing

Many names throughout the package use an arobase (`@`) to avoid name conflict with user-defined names. To simplify the compilation of the documentation, we simply make it a regular character in all the rest.

```
13 \makeatletter
```

`\axp@appendix` We setup the processing of options using `keyval` facilities; the only declared options is named `appendix`, with a default value of `append`:

```
14 \SetupKeyvalOptions{
15   family=axp,
16   prefix=axp@
17 }
18 \DeclareStringOption[append]{appendix}
19 \ProcessLocalKeyvalOptions*
```

We check that the value of the `appendix` option is valid, and add a message to the compilation log.

```
20 \ifthenelse{\equal{\axp@appendix}{append}}{
21   \message{apxproof: Appendix material appended to the document}
22 }{\ifthenelse{\equal{\axp@appendix}{strip}}{
23   \message{apxproof: Appendix material stripped}
24 }{\ifthenelse{\equal{\axp@appendix}{inline}}{
25   \message{apxproof: Appendix material inlined within the document}
26 }{
27   \errmessage{Error: unsupported option appendix=\axp@appendix for
28   package apxproof}
29 }}}}
```

2.3 Macros Common to All Compilation Modes

`\newtheoremrep` We define the high-level `\newtheoremrep` to have the same syntax as `amsthm`'s `\newtheorem`. For this purpose, we need a little trick to deal with the second optional argument, which is what `\@oparg` is used for. `\axp@newtheoremrep` is defined differently depending on the compilation mode

```
30 \newcommand\newtheoremrep[1]{%
31   \@oparg{\axp@newtheoremrep{#1}}[]%
32 }
```

`proofsketch` Simple `proofsketch` environment.

```
33 \newenvironment{proofsketch}
34   {\vskip3pt\noindent\emph{Proof sketch.} }
35   {\hfill\qedsymbol\vskip3pt}
```

`\thmhead` We redefine AMS- \LaTeX 's `\thmhead` to use a format where the repeated version of a theorem, using a theorem note, can look exactly like the original version of the theorem and its theorem counter.

```
36 \def\thmhead#1#2#3{%
37   \thmname{#1}\thmnumber{\@ifnotempty{#1}{ }\@upn{#2}}%
38   \thmnote{ #3}}
```

`\appendixrefname` We provide sensible defaults for these three user-customizable macros. Even though they are only useful in append mode, we define them for all modes so that a `\renewcommand` works in all cases.

```
39 \newcommand{\appendixrefname}{References for the Appendix}
40 \newcommand{\appendixbibliographystyle}{alpha}
41 \newcommand{\appendixbibliographyprelim}{}
```

Finally, some class-specific behavior so that theorems created by `\newtheoremrep` appear with the correct style. For now, only the styling for ACM document classes (e.g., `sig-alternate`) needs to be adapted.

```
42 \ifdefined\@acmtitlebox
43   \newtheoremstyle{mystyle}
44     {3pt}
45     {3pt}
46     {\itshape}
47     {}
48     {\scshape}
49     {.}
50     {.5em}
51     {}
52   \theoremstyle{mystyle}
53 \fi
```

2.4 Inline Compilation Mode

```
54 \ifthenelse{\equal{\axp@appendix}{inline}}{
```

`\axp@newtheoremrep` In inline mode, `\axp@newtheoremrep` undefines the existing theorem environment if it has already been defined (e.g., by the document class), invokes `\newtheorem` and creates a repeated theorem environment that behaves exactly as the regular theorem environment.

```

55 \def\axp@newtheoremrep#1[#2]#3{%
56   \expandafter\let\csname #1\endcsname\undefined
57   \expandafter\let\csname c@#1\endcsname\undefined
58   \newtheorem{#1}[#2]{#3}%
59   \NewEnviron{#1rep}[1][]{%
60     \begin{#1}[##1]\BODY\end{#1}%
61   }
62 }
```

`inlineproof` In inline mode, these environments behave like the regular `proof` environment.

```

nestedproof 63 \let\inlineproof\proof
appendixproof 64 \let\endinlineproof\endproof
65 \let\nestedproof\proof
66 \let\endnestedproof\endproof
67 \let\appendixproof\proof
68 \let\endappendixproof\endproof
```

`toappendix` In inline mode, this environment and these macros are no-ops.

```

\noproofinappendix 69 \newenvironment{toappendix}{}{}
\nosectionappendix 70 \let\noproofinappendix\relax
71 \let\nosectionappendix\relax
72 }
```

2.5 Append or Strip Compilation Modes

73 {

We now deal with the case where `apxproof` really does something useful: either append the appendix material to the document, or strip it entirely.

2.5.1 Auxiliary File for the Appendix

`\axp@proofsfile` We open a new auxiliary file, with extension `.axp`, where the appendix material will be dumped.

```

74 \newwrite\axp@proofsfile
75 \immediate\openout\axp@proofsfile=\jobname.axp
```

`proof` At the beginning of this file, we make `@` a regular character (since it will be used in several places for internal names) and reestablish the original definition of the `proof` environment and the `\section` macro.

```

76 \immediate\write\axp@proofsfile{%
77   \noexpand\makeatletter
78   \noexpand\let\noexpand\proof\noexpand\axp@oldproof
79   \noexpand\let\noexpand\endproof\noexpand\endaxp@oldproof
80   \noexpand\let\noexpand\section\noexpand\axp@oldsection
81 }
```

`\FVB@VerbatimOut` We modify the internal behavior of the `fancyvrb` package to write to the
`\FVE@VerbatimOut` `\axp@proofsfile`, without closing it and re-opening it on every write.

```

82 \def\FVB@VerbatimOut{%
83   \@bsphack
84   \begingroup
85     \FV@UseKeyValues
86     \FV@DefineWhiteSpace
87     \def\FV@Space{\space}%
88     \FV@DefineTabOut
89     \def\FV@ProcessLine{\immediate\write\axp@proofsfile}%
90     \let\FV@FontScanPrep\relax
91     \let\@noligs\relax
92     \FV@Scan}
93 \def\FVE@VerbatimOut{\endgroup\@esphack}

```

`toappendix` The entire content of this environment is put in appendix, possibly after starting a new appendix section if needed.

```

94 \newenvironment{toappendix}
95   {\axp@writesection\VerbatimOut}
96   {\endVerbatimOut}

```

2.5.2 Definition of New Theorems

`axp@seenreptheorem` Used to indicate whether a repeated theorem was just typeset, without its proof.

```

97 \newtoggle{axp@seenreptheorem}

```

`axp@rpcounter` Sequentially incremented for every repeated theorem, used to create labels.

```

98 \newcounter{axp@rpcounter}

```

`axp@newtheoremrep` When called with first argument `foobar`, we first undefine the existing `foobar` environment (and its counter) if it has already been defined (e.g., by the document class), then invoke `\newtheorem` for the regular version of the theorem `foobar` and `\newtheorem*` for an internal version `axp@foobarrp` that will be used in the appendix to restate the existing theorem.

```

99 \def\axp@newtheoremrep#1[#2]#3{%
100   \expandafter\let\csname #1\endcsname\undefined
101   \expandafter\let\csname c@#1\endcsname\undefined
102   \newtheorem{#1}[#2]{#3}%
103   \newtheorem*{axp@#1rp}{#3}%

```

We then define a `foobarrp` environment that increments the `axp@rpcounter` and typeset the regular `foobar` theorem with a label derived from the counter.

```

104 \NewEnviron{#1rep}[1][]{%
105   \addtocounter{axp@rpcounter}{1}%
106   \begin{#1}[##1]\label{axp@r\roman{axp@rpcounter}}\BODY\end{#1}%

```

We set the `axp@seenreptheorem` toggle to indicate that we are looking for the proof of the theorem, then store in a macro the content of the theorem's body.

```

107 \global\toggletrue{axp@seenreptheorem}%

```

```

108     \global\expandafter\let\csname rplet\roman{axp@rpcounter}\endcsname
109     \BODY

```

Possibly after starting a new appendix section if needed, we typeset a repeated version of the theorem using the `axp@foobarrp` environment and a reference to the previously defined label. We ignore any use of `\label` in this environment to avoid multiply defined labels.

```

110     \axp@writesection%
111     \immediate\write\axp@proofsfile{%
112         \noexpand\begin{axp@#1rp}
113         [\noexpand\ref{axp@r\roman{axp@rpcounter}}]\@ifnotempty{##1}{\noexpand##1}]%
114         \noexpand\let\noexpand\label\noexpand@gobble%
115         \expandafter\noexpand\csname rplet\roman{axp@rpcounter}\endcsname
116         \noexpand\end{axp@#1rp}
117     }
118 }
119 }

```

2.5.3 Proof Environments

`axp@oldproof` We save the definition of the existing `proof` environment.

```

120     \let\axp@oldproof\proof
121     \let\endaxp@oldproof\endproof

```

`\noproofinappendix` Utility macro that toggles `axp@seenreptheorem` to false.

```

122     \newcommand\noproofinappendix{%
123         \global\togglefalse{axp@seenreptheorem}%
124     }

```

`appendixproof` We dump the content of this in appendix, within an original `proof` environment, possibly after creating a new appendix section.

```

125     \newenvironment{appendixproof}
126     {%
127         \axp@writesection
128         \immediate\write\axp@proofsfile{%
129             \noexpand\begin{axp@oldproof}%
130         }%
131         \VerbatimOut
132     }
133     {%
134         \endVerbatimOut
135         \immediate\write\axp@proofsfile{%
136             \noexpand\end{axp@oldproof}%
137         }%
138         \noproofinappendix
139     }

```

`proof` This environment either puts the proof in appendix, if we are after a repeated theorem without its proof, or inlines it otherwise.

```

140 \renewenvironment{proof}
141   {%
142     \iftoggle{axp@seenreptheorem}{%
143       \appendixproof
144     }{%
145       \axp@oldproof
146     }%
147   }
148   {%
149     \iftoggle{axp@seenreptheorem}{%
150       \endappendixproof
151     }{%
152       \endaxp@oldproof
153     }%
154   }

```

inlineproof These two environments are synonyms for the original **proof** environment.

```

nestedproof 155 \let\inlineproof\axp@oldproof
156 \let\endinlineproof\endaxp@oldproof
157 \let\nestedproof\axp@oldproof
158 \let\endnestedproof\endaxp@oldproof

```

2.5.4 Section Management

axp@seccounter Sequentially incremented for every section, used to create labels.

```

159 \newcounter{axp@seccounter}

```

\axp@sectitle Saves the title of the last encountered section.

```

160 \def\axp@sectitle{}

```

\axp@oldsection We redefine the **\section** command to create a label based on **axp@seccounter** and to store its title in **\axp@sectitle**. Two definitions are necessary to cover the starred and unstarred use of **\section**, though most likely the former is not going to be used (since no section number will appear to refer to that section in the appendix).

```

161 \let\axp@oldsection\section
162 \def\section{\@ifstar\@section\@@section}
163 \def\@section#1{%
164   \global\edef\axp@sectitle{#1}%
165   \axp@oldsection*{#1}%
166   \addtocounter{axp@seccounter}{1}%
167   \label{axp@s\roman{axp@seccounter}}%
168 }
169 \def\@@section#1{%
170   \global\edef\axp@sectitle{#1}%
171   \axp@oldsection{#1}%
172   \addtocounter{axp@seccounter}{1}%
173   \label{axp@s\roman{axp@seccounter}}%
174 }

```

`\nosectionappendix` We remove the current section title, to indicate no section should be created in the appendix.

```
175 \newcommand{\nosectionappendix}{
176   \global\def\axp@sectitle{}%
177 }
```

`\axp@writesection` If `\axp@sectitle` is not empty, we create a new section in the appendix, referring to the main text section.

```
178 \newcommand\axp@writesection{%
179   \ifx\axp@sectitle\@empty
180   \else
181     \immediate\write\axp@proofsfile{%
182       \noexpand\def\noexpand\axp@tmp{\noexpand\ref{axp@s\roman{axp@seccounter}}}%
183       \noexpand\axp@oldsection{%
184         Proofs for Section\noexpand~\noexpand\protect\noexpand\axp@tmp{
185           (\axp@sectitle)%
186         }%
187       }%
188       \nosectionappendix
189     \fi
190   }
```

2.5.5 Append Compilation Mode

```
191 \ifthenelse{\equal{\axp@appendix}{append}}{
```

`\axp@oldbibliography` Thanks to `bibunits`'s `\defaultbibliography` macro, we set the appendix bibliography source to be the same as that of the main text.

`\bibliography`

```
192 \let\axp@oldbibliography\bibliography
193 \renewcommand\bibliography[1]{%
194   \defaultbibliography{#1}%
195   \axp@oldbibliography{#1}%
196 }
```

After the end of the main text, we add the appendix (on a new page, set in single-column mode) within a `bibunit` environment so as to typeset a separate bibliography for the appendix.

```
197 \AtEndDocument{
198   \clearpage\onecolumn\appendix
199   \begin{bibunit}[\appendixbibliographystyle]
200     \immediate\closeout\axp@proofsfile
201     \input{\jobname.axp}
202     \renewcommand{\refname}{\appendixrefname}
203     \appendixbibliographyprelim
204     \putbib
205   \end{bibunit}
206 }
207 }
```


2.5.6 Class-Specific Behavior

We conclude with some class-specific behavior.

```
\@getcl@ss We first use a little trick to store the current document class in macro \@currentclass,
\@getclass from http://tex.stackexchange.com/a/43541.
\@currentclass 208 \def\@getcl@ss#1.cls#2\relax{\def\@currentclass{#1}}
209 \def\@getclass{\expandafter\@getcl@ss\@filelist\relax}
210 \@getclass
```

ACM Document Classes

```
211 \ifdefined\@acmttitlebox

\thebibliography The section title of the bibliography is in uppercase in these document classes. In
\refname addition, the \thebibliography macro hard-codes twice the section title, so we
\appendixrefname un-hardcode it so that it can be modified in the appendix.

212 \patchcmd{\thebibliography}{References}{\protect\refname}{}{}
213 \patchcmd{\thebibliography}{References}{\protect\refname}{}{}
214 \newcommand{\refname}{REFERENCES}
215 \renewcommand{\appendixrefname}{REFERENCES FOR THE APPENDIX}

\section These document classes redefine \section in a weird way, adding the possibility
\@@section of an optional argument. We redefine them in a sane way.

216 \def\section{\@ifstar\@section{\@dblarg{\@@section}}}
217 \def\@@section[#1]#2{%
218 \global\edef\axp@sectitle{#2}%
219 \axp@oldsection{#2}%
220 \addtocounter{axp@seccounter}{1}%
221 \label{axp@s\roman{axp@seccounter}}%
222 }

223 \fi
```

lipics

```
224 \ifthenelse{\equal{\@currentclass}{lipics}}{
```

\appendixbibliographyprelim The default bibliography in the lipics document class formatting is not compatible with the alpha bibliography style. We fix this here.

```
225 \renewcommand{\appendixbibliographyprelim}{%
226 \global\let\@oldbiblabel\@biblabel
227 \def\@biblabel{\hspace*{-2em}\small\@oldbiblabel}%
228 }

229 }
230 }
```

Change History

v1.0.0-dev
 General: Initial version 1

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols			
\@@section	161, 216	\axp@proofsfile	74, 76, 89, 111, 128, 135, 181, 200
\@acmtitlebox	42, 211	\axp@rpcounter	98
\@biblabel	226, 227	\axp@seccounter	159
\@bsphack	83	\axp@sectitle	160, 164, 170, 176, 179, 185, 218
\@currentclass	208, 224	\axp@seenreptheorem	97
\@dblarg	216	\axp@tmp	182, 184
\@esphack	93	\axp@writesection	95, 110, 127, 178
\@filelist	209	B	
\@getcl@ss	208	\bibliography	192
\@getclass	208	\BODY	60, 106, 109
\@gobble	114	C	
\@ifpackageloaded	7	\clearpage	198
\@ifstar	162, 216	\closeout	200
\@noligs	91	D	
\@oldbiblabel	226, 227	\DeclareStringOption	18
\@oparg	31	\defaultbibliography	194
\@section	161, 216	E	
\@upn	37	\emph	34
A		\endappendixproof	68, 150
\appendix	198	\endaxp@oldproof	79, 121, 152, 156, 158
\appendixbibliographyprelim	39, 203, 225	\endinlineproof	64, 156
\appendixbibliographystyle	39, 199	\endnestedproof	66, 158
\appendixproof	67, 143	\endproof	10, 64, 66, 68, 79, 121
appendixproof (environment)	63, 125	\endVerbatimOut	96, 134
\appendixrefname	39, 202, 212	environments:	
\AtEndDocument	197	appendixproof	63, 125
\axp@appendix	14, 20, 22, 24, 27, 54, 191	axp@oldproof	120
\axp@newtheoremrep	31, 55, 99, 99	inlineproof	63, 155
\axp@oldbibliography	192	nestedproof	63, 155
\axp@oldproof	78, 120, 145, 155, 157	proof	76, 140
axp@oldproof (environment)	120		
\axp@oldsection	80, 161, 183, 219		

proofsketch	33		
toappendix	69, 94		
\errmessage	27		
F			
\FV@DefineTabOut	88		
\FV@DefineWhiteSpace	86		
\FV@FontScanPrep	90		
\FV@ProcessLine	89		
\FV@Scan	92		
\FV@Space	87		
\FV@UseKeyValues	85		
\FVB@VerbatimOut	82		
\FVE@VerbatimOut	82		
H			
\hfill	35		
\hspace	227		
I			
\inlineproof	63, 155		
inlineproof (environment)	63, 155		
\input	201		
\itshape	46		
L			
\label	106, 114, 167, 173, 221		
M			
\message	21, 23, 25		
N			
\nestedproof	65, 157		
nestedproof (environment)	63, 155		
\newtheorem	58, 102, 103		
\newtheoremrep	30		
\newtheoremstyle	43		
\noindent	34		
\noproofinappendix	69, 122, 138		
\nosectionappendix	69, 175, 188		
O			
\onecolumn	198		
\openout	75		
P			
\ProcessLocalKeyvalOptions	19		
\proof	9, 63, 65, 67, 78, 120		
proof (environment)	76, 140		
proofsketch (environment)	33		
\putbib	204		
Q			
\qed	35		
R			
\ref	113, 182		
\refname	202, 212		
S			
\scshape	48		
\section	76, 161, 216		
\SetupKeyvalOptions	14		
\small	227		
T			
\thebibliography	212		
\theoremstyle	52		
\thmhead	36		
\thmname	37		
\thmnote	38		
\thmnumber	37		
toappendix (environment)	69, 94		
\togglefalse	123		
\toggletrue	107		
V			
\VerbatimOut	95, 131		
\vskip	34, 35		
W			
\write	76, 89, 111, 128, 135, 181		