Using the `amsthm` Package

Version 2.20.2, April 2015
Publications Technical Group
American Mathematical Society

Contents

1   Introduction

The `amsthm` package provides an enhanced version of LaTeX's `\newtheorem`
command for defining theorem-like environments. The enhanced `\newtheorem`
recognizes a `\theoremstyle` specification (as in Mittelbach's `theorem` package)
and has a `*` form for defining unnumbered environments. The `amsthm` package
also defines a `proof` environment that automatically adds a QED symbol at
the end. AMS document classes incorporate the `amsthm` package, so everything
described here applies to them as well.

As part of the AMS LaTeX support environment, `amsthm` follows AMS style.
This differs in some respects from the style provided by the base LaTeX classes
and the package mentioned above; where a difference is known to be present,
it will be noted in these instructions. One notable difference is that, after the
end of a theorem or proof, the following line of text is assumed to start a new
paragraph, and will always be indented, whether or not a blank line or `\par` is
present.

If the `amsthm` package is used with a non-AMS document class and with the
`amsmath` package, `amsthm` must be loaded *after* `amsmath`, not before.

Since `amsthm` follows AMS publication style, formats that diverge signifi-
cantly from that style may not be able to be accommodated. Other theorem
packages *do* exist, and users with incompatible requirements who are publish-
ing elsewhere are advised to look for other possibilities on CTAN. However,
authors preparing files for publication by the AMS are expected to follow AMS
style.

This manual describes the features provided by `amsthm`. Examples are given
separately in the file `thmtest.tex`. For best understanding, examine the output
side-by-side with the input.

2   Setup and input of theorem elements

In mathematical research articles and books, theorems and proofs are among
the most common elements, but authors also use many others that fall in the
same general class: lemmas, propositions, axioms, corollaries, conjectures, def-
initions, remarks, cases, steps, and so forth (see page 7). It is natural to handle
these elements as LaTeX environments, but document classes do not provide pre-
defined environments for theorem-like elements because (a) that would make it
difficult for authors to exercise the necessary control over the automatic num-
bering, and (b) the variety of such elements is so wide that it's not possible for
a document class to provide every one that will ever be needed. Instead there

is a command `\newtheorem`, similar in effect to `\newenvironment`, that makes it easy for authors to set up the elements required for a particular document.

The `\newtheorem` command has two mandatory arguments: the first one is the environment name that the author would like to use for this element; the second one is the heading text. For example,

```
\newtheorem{lem}{Lemma}
```

means that instances in the document of

```
\begin{lem} Text text ... \end{lem}
```

will produce

> **Lemma 1.** *Text text ...*

where the heading consists of the specified text "Lemma" and an automatically generated number and punctuation. Note that there are no fixed text assignments as there are with, e.g., `\chaptername` (usually "Chapter" or "Appendix", depending on context); the heading text in the output will be exactly what you specify in the input.

If `\newtheorem*` is used instead of `\newtheorem` in this example, numbers will not be generated automatically for any of the lemmas in the document. This form of the command can be useful even if you have only one lemma and don't want it to be numbered; more often, though, it is used to produce a special named variant of one of the common theorem types. For example, if you have a lemma that should get the heading "Klein's Lemma", then the statement

```
\newtheorem*{KL}{Klein's Lemma}
```

would allow you to write

```
\begin{KL} Text text ... \end{KL}
```

and get the desired output.

Sometimes additional information is desired in the heading of a theorem or lemma, often because it is cited from another source. An optional argument is used to provide this information:

```
\begin{lem}[Alinhac-Lerner \cite{a-l}]
```

yielding the output

> **Lemma 2.** (Alinhac-Lerner [4]). *Text text . . .*

If the citation includes page numbers, the brackets in the reference must be "hidden" to prevent the closing bracket from being interpreted as the end of the optional argument to the heading:

```
\begin{lem}[Alinhac-Lerner {\cite[p.~37]{a-l}}]
```

with the output

> **Lemma 2.** (Alinhac-Lerner [4, p. 37]). *Text text . . .*

Often, theorems are referred to elsewhere in a document. The usual LaTeX `\label`-`\ref` mechanism works for this purpose. However, problems can arise when the `\label` is separated from the heading. For this reason, it is recommended to place the label immediately after the `\begin{...}[...]` command; placing it on a separate line should not cause any adverse effects.

2.1   Theorems beginning with lists

Sometimes it's desired to state a theorem in discrete steps, using a list. Preceding the list by text is recommended, to avoid problems both with linking (using `enumerate`) and with unattractive spacing.

If no text is provided, the list will follow the heading on the same line. This has three unwanted effects: the horizontal space between the theorem head and the first item will be unacceptably large, a hyperlink to the theorem will not be "live" in the PDF file (although it will be marked), and the extra vertical space that should follow the end of the list will be absent.

The best way to avoid these problems is to allow the list to start on a new line. One way to accomplish this is to follow the theorem head (and `\label`, if present) by the command `\leavevmode`. (For more information, see section 4.3.1.) However, if the theorem comes near the bottom of a page, the list might move to the top of the next page, leaving an orphaned heading. If this happens, it must be addressed as an exception, to be taken care of after the text is final; at that point, the recommended fix is to call on the `needspace` [NDS] package, or insert an explicit page break.

An alternative method of starting a new line after the heading is to provide a `\newtheoremstyle{break}`; the definition is given below on page 9. Like the `\leavevmode` approach, a `break` theorem environment is not perfect; known limitations accompany the definition.

If the list *must* start on the same line as the heading, enter the first item identifier manually, with adjustments: enter an ordinary space between the heading and the identifier, reset the left margin if the item text will require more than one line, and reset the item counter (for `enumerate`) so that it will start with 2 (or whatever is appropriate).

```
\begin{thm}[⟨optional modifier⟩]
\hangindent\leftmargini  % for a multi-line item
\label{thm:xxx}
\textup{(1)} First item ...
\begin{enumerate}
\setcounter{enumi}{1}
\item ...
\item ...
\end{enumerate}
\end{theorem}
```

The style shown here for the item identifier matches the default for AMS document classes. It may be different for other document classes; `amsthm` retains the class default. (`\textup` should be applied to hand-coded item identifiers when using an AMS class; AMS style conforms to traditional math typography in this respect, with item identifiers, cross-references, and similar elements consistently set upright, even in italic environments. However, an author should always try to match the style of the document class being used.) To match the default style of a first-level item counter when using an AMS document class, a reference may be entered with `\eqref`; however, beware if a package such as `cleveref` is being used, as it may associate the wrong element type.

An initial item input in this manner cannot be linked using a `\label`, since it is not associated with a counter; the `\label` in the above example refers to the theorem element, not to the item in the list. This is an important concept: a `\label` will always refer to the value of the most recent counter that is automatically incremented; this counter will also be the anchor for a hyperlink if the `hyperref` package is used.

3 Theorem numbering

In addition to the two mandatory arguments, `\newtheorem` has two mutually exclusive optional arguments. These govern the sequencing and hierarchy of the numbering. The numbering mechanism can be thought of this way:

```
\newtheorem{⟨env name⟩}{⟨text⟩}[⟨parent counter⟩]
\newtheorem{⟨env name⟩}[⟨shared counter⟩]{⟨text⟩}
```

The ⟨*parent counter*⟩ is comparable to `\numberwithin`; that is, numbering will restart whenever that sectional level is encountered. If a ⟨*shared counter*⟩ is specified, numbering will progress sequentially for all theorem elements using this counter. See below for a more detailed explanation.

By default each kind of theorem-like environment is numbered independently. Thus if you have three lemmas and two theorems interspersed, they will be numbered something like this: Lemma 1, Lemma 2, Theorem 1, Lemma 3, Theorem 2. If you want lemmas and theorems to share the same numbering sequence—Lemma 1, Lemma 2, Theorem 3, Lemma 4, Theorem 5—then you should use the ⟨*shared counter*⟩ to indicate the desired relationship as follows:

```
\newtheorem{thm}{Theorem}
\newtheorem{lem}[thm]{Lemma}
```

The optional argument `[thm]` in the second statement means that the `lem` environment should share the `thm` numbering sequence instead of having its own independent sequence.

To have a theorem environment numbered subordinately within a sectional unit—e.g., to get propositions numbered Proposition 2.1, Proposition 2.2, and so on in Section 2—put the name of the parent unit in square brackets in final position:

```
\newtheorem{prop}{Proposition}[section]
```

With the optional argument `[section]`, the `prop` counter will be reset to 0 whenever the parent counter `section` is incremented, and the proposition heading will have the section number prepended.

If any theorem elements are numbered by section, and (in a book) the first such element in a chapter comes before the first section, numbering will continue from the previous chapter. In such a case, reset the counter by invoking this command before the affected element:

```
\setcounter{thm}{0}
```

Note that theorem numbering is not accomplished by the same method as the `\numberwithin` facility of `amsmath`, so an attempt to use that to relate theorem numbers to equation numbers will not work in the expected way. A method for accomplishing that, as well as some other variations, is given in the AMS Author FAQ [AF], in the section on "Theorems in AMS-LaTeX".

3.1   Related topics addressed in the AMS author FAQ

- "Number" theorems with letters instead of numbers

- Restart numbering for each chapter, but don't include chapter number as part of theorem number

- Number theorems by section when chapters have sections, but by chapter only when a chapter has no sections

- Number theorems and equations with the same numbering sequence

4   Changing styles for theorem-like environments

4.1   The \theoremstyle command

The `amsthm` package supports the notion of a current theorem style, which determines what will be produced by a given `\newtheorem` command. The three theorem styles provided—`plain`, `definition`, and `remark`—specify different degrees of visual emphasis corresponding to their relative importance. The details of this typographical treatment may vary depending on the document class, but typically the `plain` style produces italic body text, while the other two styles produce roman body text. These default settings are provided:

- `plain` : italic text, extra space above and below;

- `definition` : upright text, extra space above and below;

- `remark` : upright text, no extra space above or below.

If no `\theoremstyle` command is given, the style used will be `plain`. To specify different styles, divide your `\newtheorem` commands into groups and preface each group with the appropriate `\theoremstyle`. Some examples:

```
\theoremstyle{plain}% default
\newtheorem{thm}{Theorem}[section]
\newtheorem{lem}[thm]{Lemma}
\newtheorem{prop}[thm]{Proposition}
\newtheorem*{cor}{Corollary}
\newtheorem*{KL}{Klein's Lemma}

\theoremstyle{definition}
\newtheorem{defn}{Definition}[section]
```

```
\newtheorem{exmp}{Example}[section]
\newtheorem{xca}[exmp]{Exercise}

\theoremstyle{remark}
\newtheorem*{rem}{Remark}
\newtheorem*{note}{Note}
\newtheorem{case}{Case}
```

The following list summarizes the types of structures which are normally associated with each theorem style.

`plain`	Theorem, Lemma, Corollary, Proposition, Conjecture, Criterion, Assertion
`definition`	Definition, Condition, Problem, Example, Exercise, Algorithm, Question, Axiom, Property, Assumption, Hypothesis
`remark`	Remark, Note, Notation, Claim, Summary, Acknowledgment, Case, Conclusion

4.2   Number swapping

A common style variation is to have the theorem number at the beginning of the heading instead of at the end, for example "1.4 Theorem" instead of "Theorem 1.4". As this variation is usually applied to all \theoremstyles, it is done by placing a \swapnumbers command at the beginning of the list of \newtheorem statements that should be affected. For example:

```
\swapnumbers
\theoremstyle{plain}
\newtheorem{thm}{Theorem}[section]
\theoremstyle{remark}
\newtheorem{rem}{Remark}
```

When `amsthm` is used with a non-AMS document class, and numbers are swapped, no period is set following the number; this differs from the "basic" LATEX style. A workaround will be given in the AMS Author FAQ [AF].

In some AMS document classes, the style of the swapped numbers matches that of section headings; this may not be the same style as the rest of the theorem heading.

4.3   New theorem styles

The `amsthm` package provides a `\newtheoremstyle` command to aid in the creation of custom styles, should the three predefined styles prove insufficient. The following example illustrates the use of the `\newtheoremstyle` command.

```
\newtheoremstyle{note}%    ⟨name⟩
  {3pt}%         ⟨Space above⟩¹
  {3pt}%         ⟨Space below⟩¹
  {}%            ⟨Body font⟩
  {}%            ⟨Indent amount⟩²
  {\itshape}%    ⟨Theorem head font⟩
  {:}%           ⟨Punctuation after theorem head⟩
  {.5em}%        ⟨Space after theorem head⟩³
  {}%            ⟨Theorem head spec (can be left empty, meaning 'normal')⟩
```

[1]Space above and below: these are equal, but the defaults differ when using an AMS document class vs. `\usepackage{amsthm}`. For AMS classes, this is `.5\baselineskip±.2\baselineskip`; for `amsthm` used as a separate package, it is the local value of `\topsep`. If these arguments are left empty, the current defaults are used.
[2]Indent amount: empty = no indent, `\parindent` = normal paragraph indent
[3]Space after theorem head: `{ }` = normal interword space; `\newline` = line break

Apply the new theorem style in the same manner as the predefined ones. Here are some styles that have been requested by authors.

4.3.1   Theorem style `break`

This style will break after the theorem heading and start a new line.

```
\newtheoremstyle{break}%
  {}{}%
  {\itshape}{}%
  {\bfseries}{}%  % Note that final punctuation is omitted.
  {\newline}{}
```

This style can be used for a theorem beginning with a list. When used with **enumerate** and an AMS document class, all items are properly labeled and will link. However, the vertical spacing needs help; a conflict between definitions prevents the first item from starting on a new line—it looks almost the same as a default theorem beginning with an enumerated list. To repair this problem, begin the theorem like this:

```
\begin{breakthm}[...]
\leavevmode \vspace{-\baselineskip}
\begin{enumerate}
```

`\leavevmode` alone will leave a full blank line after the theorem head.

One more problem may arise: if the theorem starts close to the end of a page, the list could be split to a new page, leaving an orphaned heading. Make a note to address this when the text is final; then call on the `needspace` [NDS] package, or insert an explicit page break.

4.3.2   Theorem with bold note field

If the note—with surrounding parentheses—is particularly important, setting the note field in bold may be desired. This variation numbers the theorem, an option not available when using `\newtheorem*`.

```
\newtheoremstyle{bfnote}%
  {}{}%
  {\itshape}{}%
  {\bfseries}{.}%
  { }%
  {\thmname{#1}\thmnumber{ #2}\thmnote{ (#3)}}
```

Parentheses may be omitted from around the note using this trivial adaptation of the formulation with a bold note.

```
\newtheoremstyle{noparens}%
  {}{}%
  {\itshape}{}%
  {\bfseries}{.}%
  { }%
  {\thmname{#1}\thmnumber{ #2}\thmnote{ #3}}
```

4.3.3   Other styles

For examples, see the file `thmtest.tex`.

5 Proofs

A predefined `proof` environment provided by the `amsthm` package produces the heading "Proof" with appropriate spacing and punctuation. The proof environment is primarily intended for short proofs, no more than a page or two in length; longer proofs are usually better done as a separate `\section` or `\subsection` in your document.

5.1 The beginning of a proof

An optional argument of the proof environment allows you to substitute a different name for the standard "Proof". If you want the proof heading to be, say, "Proof of the Main Theorem", then write

```
\begin{proof}[Proof of the Main Theorem]
```

Because of the way the `proof` environment is defined (as a LATEX list), a substitute "name" that is longer than one output line will not break as it should; this is on the list for repair. In the meantime, this redefinition in your preamble will accomplish the desired result.

```
\makeatletter
\renewenvironment{proof}[1][\proofname]{\par
  \pushQED{\qed}%
  \normalfont \topsep6\p@\@plus6\p@\relax
  \trivlist
  \item\relax
        {\itshape
      #1\@addpunct{.}}\hspace\labelsep\ignorespaces
}{%
  \popQED\endtrivlist\@endpefalse
}
\makeatother
```

A proof beginning with a list can be approached in the same manner as a theorem; see section 2.1.

5.2 The end of a proof

A QED symbol, □, is automatically appended at the end of a `proof` environment. To substitute a different end-of-proof symbol, use `\renewcommand` to redefine the command `\qedsymbol`. For a long proof done as a subsection or

section instead of with the `proof` environment, you can obtain the symbol and the usual amount of preceding space by using `\qed`; this will work properly only if it follows ordinary text.

A blank line preceding `\end{proof}` (or before an explicit `\qed`) disables the mechanism that places the QED symbol, and may even allow that symbol to be set at the top of a new page.

Placement of the QED symbol can be problematic if the last part of a `proof` environment is a displayed equation or list environment or something of that nature. In that case put a `\qedhere` command at the place where the QED symbol should appear, for example,

```
\begin{proof}
...
\begin{equation}
G(t)=L\gamma!\,t^{-\gamma}+t^{-\delta}\eta(t) \qedhere
\end{equation}
\end{proof}
```

When used with the `amsmath` package, version 2 or later, `\qedhere` will position the QED symbol flush right; with earlier versions, the symbol will be spaced a quad away from the end of the text or display.

If `\qedhere` produces an error message in an equation, try using `\mbox{\qedhere}` instead.

`\qedhere` does not work with `eqnarray` or with equations input with `$$`; the display environments defined by `amsmath` and the `\[ ... \]` form of un-numbered equations have been specially crafted for this purpose.

When a proof ends with a list environment, place `\qedhere` as the last thing before `\end{enumerate}` or `\end{itemize}`.

If, for some reason, it is desired to omit the QED symbol from the end of a proof, it can be suppressed by including this definition just before `\end{proof}`:

```
\renewcommand{\qedsymbol}{}
```

If the symbol is to be omitted from *all* proofs in the document, place the redefinition in the preamble.

If the QED symbol is needed outside of the `proof` environment, use `\qed` rather than `\qedhere`.

6   Known problems

The following problems are listed for attention with the next update of `amsthm`.

- Problems with `\qedhere`.

  - When `\qedhere` is used with an unstarred multi-line display structure, the final line is not numbered. (This is relevant only when numbers are on the left; `\qedhere` is inappropriate for use with displays numbered on the right.)
  - In an alignment, if no `&` is present, the box does not go flush to the right margin.
  - For a one-line equation, both the number and the box appear, but the equation is shifted left.

- Other problems inherent in `amsthm`.

  - Hyperlinking the first item when a theorem begins with a list.

- Problems of interaction with AMS document classes. Corrections will be made in the classes, not in `amsthm`.

  - In `amsbook`, when a theorem appears before a chapter's first numbered section, theorem numbering continues from the previous chapter.

7   Other packages

Some packages provide additional theorem functionality, with options that recognize the presence of `amsthm` and adjust for it.

- `mdframed` [MDF] provides the ability to set off theorems (and other elements) with frames of various shapes and colors. It should be loaded *after* `amsthm` if both are used.

- `thmtools` [THT] provides an alternate key-value syntax for theorem specification, as well as variant presentation styles, both predefined and user definable. A QED symbol can be applied to any theorem element, not just to proofs. An adaptable `\listoftheorems` is also available.

For details, refer to the package manuals. Both packages are included in TeX Live, so can be accessed with `texdoc` ⟨*package*⟩.

Some packages that offer potentially useful features have limitations that conflict with the implementation of theorems with AMS document classes or `amsthm`. Here are some we know about.

- `paralist` overrides the settings in `amsthm` that apply `\upshape` to item numbers in an `enumerate` environment within a theorem; the numbers become italic, which is undesired.

- `wrapfig` documents that it does not coexist with lists. Both theorems and proofs in `amsthm` are based on `\trivlist`. An insertion using this method may cause succeeding paragraphs to be indented by the width of the insert even if the depth of the insertion does not warrant it.

- `floatflt` and `picinpar` suffer from the same limitation.

8 Potential modifications

Feedback from users and experience with AMS publications have pointed out several areas where added flexibility can benefit both users of `amsthm.sty` and developers of "derivative" document classes based on the AMS set.

These suggestions are relatively easy to implement:

- Change hard-coded font specifications (e.g., `\itshape`) to commands (e.g., `\proofheadingfont`) to allow modification.

- Provide commands instead of hard-coded values for other elements, such as punctuation.

The following suggestions, while having merit, require more work, including redesign of existing code:

- Extend applicability of `\qedhere` to theorem-class elements as well as to proofs.

- Provide a means to ignore the definition of `proof` to avoid name conflicts with some non-AMS document classes.

- Consider key-value mechanism for specifying `\newtheoremstyle` options.

- Provide a new theorem style that begins with a list.

9 Providing feedback

Reports on bugs found in this package should be submitted to

<div align="center">

`tech-support@ams.org`

</div>

For best results, a brief, compilable example should accompany the report.

An alternate method is to use the LaTeX bugs reporting mechanism, specifying `AMS-LaTeX` as the bug category.

References

[AF]   AMS Author FAQ, `http://www.ams.org/authors/author-faq`

[MDF]   The `mdframed` package, Marco Daniel and Elke Schubert,
        2013/07/01, v1.9b,
        `http://mirror.ctan.org/macros/latex/contrib/mdframed`

[NDS]   The `needspace` package, Peter Wilson, 2010/09/12, v1.3d,
        `http://mirror.ctan.org/macros/latex/contrib/needspace`

[THT]   `Thmtools` Users' Guide, Ulrich M. Schwarz, 2014/04/21 v66,
        `http://mirror.ctan.org/macros/latex/exptl/thmtools`