

deleq – a L^AT_EX Macro for Partial Numbering of Equations*

Mats Dahlgren[†]

1997/06/05

Abstract

To enable a more flexible equation numbering, especially “partial” equation numbers (‘3a’, ‘3b’ *etc.*), the `deleq` package has been developed. It can produce partial equation numbers intermixed with ordinary equation numbers also in an `eqnarray`-like environment, the intermixing can occur within one environment. References to a partially numbered equation can be both the complete equation number (‘3b’) or only the main equation number (‘3’). Furthermore, equation numbers can be recycled without disturbing the ordinary equation numbers. The package also provides commands for putting commentary text in an `eqnarray` environment. Both standard L^AT_EX class options `leqno` and `fleqn`¹ work with `deleq`.
This file and the package: Copyright © 1994-1998 by Mats Dahlgren. All rights reserved.

1 Introduction

`deleq` is a L^AT_EX package which makes partial numbering of equations possible. It is meant to be used when numbering such as 3a, 3b, *etc.* is desired. The default is to give an equation number like ‘3a’ without period and with the ‘a’ typeset in roman font. It can be used in `article` as well as `book` and `report` document classes. The examples given below are valid for the `article` class. `deleq` is fully compatible with the `leqno` documentclass option and almost fully with the `fleqn` class option. Equations can be numbered either 3, 3a, 3b, ..., or 2, 3a, 3b, Also, equation numbers within `eqnarray`-like environments can be numbered 3, 3a, 3b, ... or starting at 3a, which can follow both after equation 2b or 3. An `eqnarray`-like environment can have its first equation numbered 3b if the nearest previous equation is 3a. Within one `eqnarray`-like environment, equations can be numbered 3a, 3b, 4, 4a, 4b, 5a, 5b, *etc.*, and also not numbered lines are possible (‘`\nonumber`’ works with some limitations). Furthermore, equation numbers can be “recycled”. If equation 3 is repeated after equation 8, it can still have the numbering set to 3, and be followed by (a new) equation 9. “Recycled” equation numbers can receive partial numbers (3a, 3b, ...); partially numbered equations can also be “recycled” (but at this stage, the latter can not receive new partial numbers neatly, it will come out like ‘3ba’ if equation 3b is the “recycled” equation). Two commands are supplied which enable the user to write commentary texts in `eqnarray`-like environments without interfering with the alignment.

This userguide is also available in .pdf-format on the internet. It is found from my L^AT_EX web page: <http://www.homenet.se/matsd/latex/>

*This document describes `deleq` version 4.41a, last revised 1997/06/05

[†]Email: matsd@sssk.se Web: <http://www.homenet.se/matsd/>

¹Not all commands of `deleq` work with the `fleqn` option.

1.1 History

`deleq.sty` was originally written for L^AT_EX 2.09, and contained several commands which became obsolete with the introduction of L^AT_EX 2_ε. These commands have been removed. The first release for L^AT_EX 2_ε was v.4.0, of Oct. 14, 1994. The December 1994 release (v.4.1, Dec. 27, 1994) was the first which was compatible with the `docstrip` utility of Johannes Braams, Denys Duchier and Frank Mittelbach. In release v.4.2 new abilities to reference entire `deqarr` and `ddeqar` environments and enhanced abilities for recycled equation numbers were introduced. Also in `deleq` v.4.2 this documentation was revised and some internal commands compacted. A previous release, v.4.3, was made necessary because the changed handling of arguments in connection with the options in L^AT_EX 2_ε of 1995/06/01. In release v.4.4, compatibility with the L^AT_EX class option `fleqn` was introduced.² In v.4.41 only some changes to the documentation have been introduced.

`deleq` version 4.41 has been tested with L^AT_EX 2_ε of 1997/06/01 running T_EX 3.14159 in MiK_TE_X 1.07 under Windows 95. Please send bug reports (see below), corrections, additions, suggestions, *etc.* to me at `matsd@sssk.se`. Version 3.02 is the last (non-supported) version for L^AT_EX 2.09. (Command names are a mixture of L^AT_EX, Swedish and a bit of the author's fantasy...)

1.2 Known Problems

- When using `\eqreqno`, `\deqreqno`, and `\ddeqreqno` on the last line in a `deqarr`, `ddeqar`, `eqnarray`, or `eqnarray*`, there will be an extra blank line with an equation number (not in the * form) at the end of the environment. To avoid this problem, use `\eqreqno[-\jotbaseline]{F00} \nonumber` instead of `\eqreqno{F00}` (the analogues for `\deqreqno` and `\ddeqreqno` are obvious).

2 Userguide

2.1 Requirements

The file `deleq.sty` must be available in the user's `TEXINPUTS` directories. It requires L^AT_EX 2_ε of 1996/12/01 (or newer).

2.2 Usage

The package is included by stating

```
\usepackage{deleq}
```

In the document preamble. The documentclass option `leqno` is fully supported, and the option `fleqn` is recognized by the environments defined in `deleq`. Other class or package options do not have any effect on `deleq`.

2.3 Environments

The package defines the following five environments:

- `deqn` `deqn` is an equation environment for automatic numbering of the first equation of type '3a';
- `ddeqn` `ddeqn` is an equation used for automatic numbering of the equations following the one in `deqn`, see also details below;
- `deqarr` `deqarr` is an environment for automatic partial equation numbering in `eqnarray`-like environment

²`fleqn` incompatibility was pointed out by Peter Kruijt (`peterk@wfw.wtb.tue.nl`)

with `deqn` functionality for the first equation;

`ddeqar` `ddeqar` similar to `deqarr` environment, but with `ddeqn` behaviour for the first equation, see details below;

`deqrarr` `deqrarr` this is an ‘`eqnarray`’-like environment specially designed for recycled equation numbers.

2.4 Commands

The package defines the following fourteen user commands:

```
\nydeqno
\heqno
\reqno{FOO}
\rndeqno{FOO}
\rdeqno{FOO}
\eqreqno[len]{FOO}
\deqreqno[len]{FOO}
\ddeqreqno[len]{FOO}
\arrlabel{FOO}
\where
\rem{text}
\nydeleqno
\deleqno
\jotbaseline
```

2.5 Notice

- | | |
|---|--|
| <code>\nydeqno</code>
<code>\heqno</code>
<code>\arrlabel</code>
<code>deqarr</code>
<code>ddeqarr</code> | <ol style="list-style-type: none">1. The use of <code>\arrlabel</code> may give unexpected results if any of the commands <code>\nydeqno</code> or <code>\heqno</code> is used in the same <code>deqarr</code> or <code>ddeqar</code> environment.2. If you use <code>\nydeqno</code>, <code>\heqno</code>, <code>\reqno</code>, <code>\rndeqno</code>, <code>\rdeqno</code>, <code>\deqreqno</code>, <code>\ddeqreqno</code>, <code>\nydeleqno</code>, or <code>\deleqno</code> in combination with <code>\label{FOO}</code>, <code>\label{FOO}</code> should appear after the <code>deleq</code>-command;3. If you change the appearance of equation numbers (<i>e.g.</i> use ‘[2]’ instead of ‘(2)’), <code>deleq</code>’s commands may not typeset equations with partial numbers like your ordinary equations.4. The class option <code>fleqn</code> is not recognized by the commands <code>\reqno</code>, <code>\rndeqno</code>, <code>\rdeqno</code>, <code>\nydeleqno</code>, and <code>\deleqno</code>. |
|---|--|

3 Syntax

Here follows a more detailed description of the different environments and commands.

- | | |
|--------------------|---|
| <code>deqn</code> | <pre>\begin{deqn} ... \end{deqn}</pre> <p>Typesets an equation just like <code>\begin{equation} ... \end{equation}</code> does, but gives it a number such as ‘3a’ instead of ‘3’ (always ‘a’). It steps the main equation counter.</p> |
| <code>ddeqn</code> | <pre>\begin{ddeqn} ... \end{ddeqn}</pre> <p>Typesets an equation just like <code>\begin{equation} ... \end{equation}</code> does, but gives it a number such as ‘3b’, ‘3c’, ... instead of ‘3’, ‘4’, ... when following after another partially</p> |

numbered equation. When following after an ordinarily numbered equation ('3') it gives a partial equation number with the last used ordinary equation number, *e.g.* '3a'. Can be made to produce the result of `deqn` environment by the use of `\nydeqno` (see below). `ddeqn` does not step the main equation counter.

`deqarr` `\begin{deqarr} ... \end{deqarr}`

Typesets an equation array just like `\begin{eqnarray} ... \end{eqnarray}` does, but gives the first numbered equation a number like '3a' instead of '3' (always 'a'), and the following '3b', '3c' *etc.* It steps the main equation counter. The `\nonumber` command works just like in the `eqnarray` environment on the partial equation counter. Thus, if you use `\nonumber` on all lines in an `deqarr` environment (or on all lines before or between `\heqno` or `\nydeleqno`) the equation numbering in your document will be incorrect. There is no warning issued by L^AT_EX if this happens. If you want blank lines and then a `\heqno`'ed equation followed by partially numbered equations, use the `ddeqar` environment instead. The same is true for `\nydeqno`'ed equations.

`ddeqar` `\begin{ddeqar} ... \end{ddeqar}`

Typesets an equation array just like `\begin{eqnarray} ... \end{eqnarray}` does, but gives the equation numbers such as '3a' if following after an equation numbered '3' (ordinary equation number) and numbers such as '3c' if following after a partially numbered equation '3b'. Can be made to produce the result of `deqarr` environment by the use of `\nydeqno`. `ddeqar` does not step the main equation counter. The `\nonumber` command works like in the `deqarr` environment.

`deqrarr` `\begin{deqrarr} ... \end{deqrarr}`

This environment is meant to host recycled equations and has `eqnarray` structure. By default, it issues no equation number at all on a line which ends with `\\`. The use of `\label{FOO}` within `deqrarr` returns the present partial equation number of recycled equations (just like with `\rndeqno` and `\rdeqno`). Note that neither `\heqno` nor `\nydeleqno` gives any equation numbers at all in the `deqrarr` environment.

`\nydeqno` `\nydeqno`

Used within `deqarr` and `ddeqar` environments to step the main equation number by one and reset the partial equation number to 'a'; thus, `\nydeqno` gives equation number '4a' when following after equation '3c'.

`\heqno` `\heqno`

Used within `deqarr` and `ddeqar` environments to step the main equation number by one and to produce an ordinary equation number; thus, gives equation number '4' when following equation '3c'. Equations following the `\heqno`-ed will be partially numbered with the `\heqno`-ed equation's number as the main number, *e.g.* '4a' (unless it has a `\nydeqno` command, which in this case would produce the equation number '5a').

`\reqno` `\reqno{FOO}`

Is used when repeating an equation with its original number, "recycling" the equation number. `\reqno` takes the argument `FOO`, which has to be defined by a `\label{FOO}` in the original equation. It can only be used within `$$... $$`. It does not affect the equation number counter, nor the ordinary partial equation number counter. However, it resets the partial equation number counter for the `\rndeqno`, `\rdeqno`, `\deqreqno`, and `\ddeqreqno` commands. If used with a `\label{FOO}` command, the `.aux`-file will only contain the page

number of label F000. This command is not compatible with the class option `fleqn`. Instead of `$$... \reqno{F00} $$`, use a one-line `deqarr` environment with the commands `\eqreqno[-\jotbaseline]{F00} \nonumber` at the end.

`\rndeqno` `\rndeqno{F00}`

Adds a partial equation number to an old equation number, specified by the F00 label. If F00 refers to equation ‘4’, `\rndeqno{F00}` will result in equation number ‘4a’ (always ‘a’). It can only be used within `$$... $$`. It does not affect the equation number counter, nor the ordinary partial equation number counter. However, it resets the partial equation number counter for the `\rndeqno`, `\rdeqno`, `\deqreqno`, and `\ddeqreqno` commands. The argument of `\rndeqno` follows the same rules as that of `\reqno`. When used with the `\label{F000}` command, a reference to F000 will only return the partial equation number (and the page number). To make a complete reference to an equation which has a `\rndeqno` command, say `\ref{F00}\ref{F000}`. This command is not compatible with the class option `fleqn`. Instead of `$$... \rndeqno{F00} $$`, use a one-line `deqarr` environment with the commands `\deqreqno[-\jotbaseline]{F00} \nonumber` at the end.

`\rdeqno` `\rdeqno{F00}`

Same as `\rndeqno{F00}` but without resetting any equation number counter and giving consecutive partial equation numbers (‘4b’, ‘4c’, *etc.*). **Note:** There is nothing preventing the repeated use of `\rndeqno{F00}` and `\rdeqno{F00}` for the same label F00. This will result in numbering such as ‘3a’ (if F00 refers to equation ‘3’) occurring several times. This command is not compatible with the class option `fleqn`. Instead of `$$... \rdeqno{F00} $$`, use a one-line `deqarr` environment with the commands `\ddeqreqno[-\jotbaseline]{F00} \nonumber` at the end.

`\eqreqno` `\eqreqno[len]{F00}`

This command is the array-like version of `\reqno{F00}` and is used in much the same way. The optional argument *len* is a length which is added between consecutive rows in the array-like structure. When using `\eqreqno` no ‘\’ should be issued at the end of the line, it is embedded in the command. This is giving a strange appearance if `\eqreqno` is used on the last line of the array-like structure, namely an extra blank line (with equation number). To avoid this problem, specify a negative length for *len*, preferably ‘`-\jotbaseline`’ (see below), and issue a ‘`\nonumber`’ afterwards.

`\deqreqno` `\deqreqno[len]{F00}`

This is the ‘`\rndeqno`’ version of ‘`\eqreqno`’ and is used as the former with the latter’s abilities at the end of a line.

`\ddeqreqno` `\ddeqreqno[len]{F00}`

This is the ‘`\rdeqno`’ version of ‘`\eqreqno`’ and is used as the former with the latter’s abilities at the end of a line.

`\arrlabel` `\arrlabel{F00}`

This command is a version of L^AT_EX’s ordinary `\label{F00}` command meant to be used in `deqarr` and `ddeqarr` environments. In these environments, a `\label{F00}` command gives a reference to the specific equation, *e.g.* 3b, whereas `\arrlabel{F00}` in the same position gives a reference containing only the main equation number, 3 in this example. For restrictions, see the section “Notice” above. The four commands `\eqreqno`, `\deqreqno`, `\ddeqreqno`, and `\arrlabel` were created after inspiration from Larry Jones (`schnuff@mit.edu`).

`\where` `\where`

The command `\where` typsets the text “where” (in the default LR-font) flush left on a separate row in `eqnarray`, `deqarr`, `ddeqar`, and `deqrarr` environments, and preserves the environment’s alignment for rows to come.

`\remtext` `\rem{text}`

The command `\rem{text}` typsets the text “*text*” (in the default LR-font) flush left on a separate row in `eqnarray`, `deqarr`, `ddeqar`, and `deqrarr` environments, and preserves the environment’s alignment for rows to come. `\where` is equivalent to `\rem{where}`.

`\nydeleqno` `\nydeleqno`

Gives a new partial equation number when used within `$$... $$`. Thus, writing `$$... \nydeleqno $$` is equivalent to writing `\begin{deqn} ... \end{deqn}`. This command is not compatible with the class option `fleqn`, use `\begin{deqn} ... \end{deqn}` instead.

`\deleqno` `\deleqno`

Gives a partial equation number when used within `$$... $$`. Thus, writing `$$... \deleqno $$` is equivalent to writing `\begin{ddeqn} ... \end{ddeqn}`. This command is not compatible with the class option `fleqn`, use `\begin{ddeqn} ... \end{ddeqn}` instead.

`\jotbaseline` `\jotbaseline`

This is a rubber length which is set to be the sum of `\baselinsekip` and `\jot` each time a `deqarr`, `ddeqar`, or a `deqrarr` environment is entered. It is meant to be used with the `\eqreqno` command and its relatives to eliminate the problems when these commands are used on the last line of the environment. (The length `\baselineskip + \jot` is the length \LaTeX skips between two empty lines in an ‘`eqnarray`’ environment and its derivatives defined in `deleq`.)

4 Example

After running \LaTeX on `deleq.ins`, there is an example available in the file `delex.tex`, which makes use of all the environments and commands defined in the `deleq` package. Typset it and see with your own eyes what the results are!

5 Sending a Bug Report

`deleq` is most likely to contain bugs. Reports of bugs in the package are most welcome. Before filing a bug report, please take the following actions:

1. Ensure your problem is not due to your inputfile;
2. Ensure your problem is not due to your own package(s) or class(es);
3. Ensure your problem is not covered in the section “Known Problems” above;
4. Try to locate the problem by writing a minimal \LaTeX input file which reproduces the problem. Include the command


```
\setcounter{errorcontextlines}{999}
```

 in your input;
5. Run your file through \LaTeX ;

6. Send a description of your problem, the input file and the log file via e-mail to:
`matsd@sssk.se`

Enjoy your L^AT_EX!

mats d.