

Numbering individual lines of equation array's

Johannes Braams
T_EXnik
Kooiensewater 62
2715 AJ Zoetermeer
The Netherlands
Internet: JLBraams@cistron.nl

1999/03/03

This package defines the `subeqnarray` and `subeqnarray*` environments, which behave like the equivalent `eqnarray` and `eqnarray*` environments, except that the individual lines are numbered like 1a, 1b, 1c, etc.

To refer to these numbers an extra label command `\slabel` has been defined.

Many of this code was taken from `latex.tex` and modified for this purpose.

1 Initial Code

`\c@subequation` We need to allocate a new counter for the `subequation` environment. It is reset by the `equation` counter.

```
1 (*package)
2 \newcounter{subequation}[equation]
```

`\thesubequation` The representation of the counter `subequation` includes the `equation` counter

```
3 \def\thesubequation{\theequation\alph{subequation}}
```

2 Option Handling

The standard L^AT_EX options `leqno` and `fleqn` are recognised by this package.

```
4 %
5 %   When \Lopt{leqno} is used the equation numbers should appear on
6 %   the left side of the equation. The numbers are generated by
7 %   |\@subeqnnum| which needs a different definition to achieve this
8 %   effect.
9 %   \begin{macrocode}
10 \DeclareOption{leqno}{%
11   \def\@subeqnnum{\hbox to .01\p@{\}\rlap{\reset@font\rmfamily
12     \hskip -\displaywidth(\thesubequation)}}}
```

The default definition of `\@subeqnnum`.

```
13 \DeclareOption{reqno}{%
14   \def\@subeqnnum{\reset@font\rmfamily (\thesubequation)}}}
```

When the option `fleqn` is used, the equations have to be printed flush left, with an indent of `\mathindent`; the equations are separated from the surrounding text by `\topsep` (plus `\partopsep` if necessary) and the width of the display is `\linewidth`.

```

15 \DeclareOption{fleqn}{%
16   \def\subeqn@start{%
17     \tabskip\mathindent
18     \abovedisplayskip\topsep
19     \ifvmode\advance\abovedisplayskip\partopsep\fi
20     \belowdisplayskip\abovedisplayskip
21     \belowdisplayshortskip\abovedisplayskip
22     \abovedisplayshortskip\abovedisplayskip
23     $$\everycr{\halign to \linewidth}}% $$

```

The default will be to have displayed equations to the width of `\displaywidth`.

```

24 \DeclareOption{deqn}{%
25   \def\subeqn@start{%
26     \tabskip@centering
27     $$\everycr{\halign to \displaywidth}}% $$

```

We don't support any other options

```

28 \DeclareOption*{\OptionNotUsed}

```

3 Executing Options

Make sure the `\@eqnnum` is defined by specifying `reqno` as a default option. Specifying `deqn` as a default option defines `\subeqn@start`.

```

29 \ExecuteOptions{reqno,deqn}

```

Now see if the use specified any options.

```

30 \ProcessOptions

```

4 The main code

`\slabel` A new label command to refer to subequations. It works like the `\label` command and was taken from `latex.tex`.

```

\slabel{F00} writes the following on file \@auxout
\newlabel{F00}{{\eval(\@currentlabel)}{\eval(\thepage)}}

```

```

31 \newcommand\slabel[1]{%
32   \@bsphack
33   \if@filesw
34     {\let\thepage\relax
35     \def\protect{\noexpand\noexpand\noexpand}%
36     \edef\@tempa{\write\@auxout{\string
37       \newlabel{#1}{{\thesubequation}{\thepage}}}}%
38     \expandafter\@tempa
39     \if@nobreak \ifvmode\nobreak\fi\fi
40   \fi\@esphack}

```

`subeqnarray` The `subeqnarray` environment steps the equation counter, sets the subequation counter equal to 1 and behaves much like the `eqnarray` environment. Note the `\@currentlabel` is defined to use the equation counter. This is done so that an

entire array can be referred to using the value of the equation counter. Hence the need for the `\slabel` command.

```

41 \newenvironment{subeqnarray}%
42   {\stepcounter{equation}%
43   \def\@currentlabel{\p@equation\theequation}%
44   \global\c@subequation\@ne
45   \global\@eqnswtrue\m@th
46   \global\@eqcnt\z@\let\\\@subeqnqr
47   \subeqnqstart
48   \bgroup\hskip\@centering
49   $\displaystyle\tabskip\z@skip{##}$\@eqnse1
50   &\global\@eqcnt\@ne \hskip \tw@\arraycolsep \hfil${##}$\hfil
51   &\global\@eqcnt\tw@ \hskip \tw@\arraycolsep
52   $\displaystyle{##}$\hfil \tabskip\@centering
53   &\global\@eqcnt\thr@@
54   \hbox to\z@\bgroup\hss##\egroup\tabskip\z@skip\cr}
55   {\@@subeqnqr\egroup $$\global\@ignoretrue}

```

`\@subeqnqr` These macros handle the user command `\\`; they are adapted from the ones used or the `eqnarray` environment.

First the presence of a `*` detected and the right penalty selected.

```

56 \def\@subeqnqr{\ifnum0='}\fi@ifstar{\global\@eqpen\@M
57   \@ysubeqnqr}{\global\@eqpen\interdisplaylinepenalty \@ysubeqnqr}}

```

`@ysubeqnqr` This macro is called by `\@subeqnqr` and checks if the user requested any extra vertical space. It calls `\@xsubeqnqr` with the wanted amount of space as its argument.

```

58 \def\@ysubeqnqr{\@ifnextchar [{\@xsubeqnqr}{\@xsubeqnqr[\z@skip]}}

```

`\@xsubeqnqr` This macro calls `\@@subeqnqr` to put in extra `&`'s if needed, generating an error if the number of columns is too large. Then the penalty selected earlier and the white space requested are inserted.

```

59 \def\@xsubeqnqr[#1]{\ifnum0='}\fi}\@subeqnqr
60   \noalign{\penalty\@eqpen\vskip\jot\vskip #1\relax}}

```

`\@@subeqnqr` Check the number of columns, and insert extra `&` if needed. If there appear to be more than 3 columns an error is signalled.

```

61 \def\@@subeqnqr{\let\@tempa\relax
62   \ifcase\@eqcnt \def\@tempa{& &}\or \def\@tempa{& &}
63   \or \def\@tempa{&}\else
64   \let\@tempa\@empty
65   \@latexerr{Too many columns in subeqnarray environment}\@ehc\fi
66   \@tempa \if@eqnsw\@subeqnnum\refstepcounter{subequation}\fi
67   \global\@eqnswtrue\global\@eqcnt\z@\cr}

```

`subeqnarray*` This environment is basically the same as the `eqnarray` environment, but it is provided just for completeness.

```

68 \newenvironment{subeqnarray*}%
69   {\def\@subeqnqr{\nonumber\@ssubeqnqr}\subeqnarray}
70   {\global\advance\c@equation\m@ne\nonumber\endsubeqnarray}

```

`\@ssubeqnrcr` This is used in the `esubqnarray*` environment.

```
71 \let\@ssubeqnrcr\@subeqnrcr
72 \end{package}
73 \end{sample}
```

5 An example of the use of this package

When you run the following document through L^AT_EX you will see the difference between the `subeqnarray` and `eqnarray` environments.

```
74 \end{sample}
75 \documentclass[fleqn]{article}
76 \usepackage{subeqnarray}
77 \begin{document}
78 This document shows an example of the use of the \emph{subeqnarray}
79 environment. Here is one:
80 \begin{subeqnarray}
81 \label{eqw}
82 \label{eq0}
83 x & = & a \times b \\
84 \label{eq1}
85 & = & z + t \\
86 \label{eq2}
87 & = & z + t
88 \end{subeqnarray}
89 The first equation is number~\ref{eq0}, the last is~\ref{eq2}. The
90 equation as a whole can be referred to as equation~\ref{eqw}.
91
92 To show that equation numbers behave normally, here's an
93 \emph{eqnarray} environment.
94 \begin{eqnarray}
95 \label{eq10}
96 x & = & a \times b \\
97 \label{eq11}
98 & = & z + t \\
99 \label{eq12}
100 & = & z + t
101 \end{eqnarray}
102
103 These are equations~\ref{eq10},~\ref{eq11} and~\ref{eq12}.
104 \end{document}
105 \end{sample}
```