Userjs Guide

American Mathematical Society
January 1995

Contents

Chapter

What is i/j, and why would anyone want to use it?

What is i/j?

The name / is used for convenience to describe a set of loosely related files that are distributed together by the American Mathematical Society. Basically they may be described as miscellaneous enhancements to LATEX/ for *superior information structure of mathematical documents* and *superior printed output*. Because / is an extension for LATEX/, which in turn is a imacro packagejmacro package for the / typesetting program, it follows that in order to use any of the pieces of / you need to have / and LATEX/ installed first.

LATEX/ by itself does a rather good job of typesetting mathematics, compared to non-/-based software; it doesnjt add much, however, to the basic set of mathematical capabilities that it adopted from the Plain / macro package.

At the same time that LATEX/ was being developed by Leslie Lamport (roughly 1982m1986), the American Mathematical Society was throwing its resources into the development of a different macro package known as /, written by Michael Spivak. By 1987 or so it became evident that / and LATEX/ had complementary feature sets: / focused on the typesetting of math formulas and on fine-tuning typically done by publishers, and was relatively weak in other areas (for example no automatic numbering or cross-reference facilities); LATEX/ focused on document structure and logical markup of text, and had a comparatively limited set of features for dealing with math formula contents. This situation led to dissatisfaction among both / and LATEX/ users who saw desirable features tantalizingly out of reach in the other macro package. So the American Mathematical Society looked into the question of producing some sort of combination of the two macro packages that would better serve mathematicians in their writing tasks. The decision that was eventually taken was to graft the mathematical capabilities of / onto the base stock of LATEX/ through an extension package: /. Most of the programming work was done by Frank Mittelbach and Rainer Schöpf in 1989m1990 and version 1.0 of / was released in mid-1990.

Why would a LATEX/ user want to bother with /?

If you are just starting out as a LATEX/ user, youjll probably have to take our word for this (or the word of friends and colleagues), but:

If your writing contains a significant proportion of mathematics, and you care about the quality of the printed results, then sooner or later youjll find shortcomings in standard LATEX/ and want to remedy them. Chances are that at least the first few of the shortcomings you encounter will be ones that are already addressed by an / package. If you want to have maximum mathematical typesetting power ready at hand, rather than stop to cast about for a solution whenever you run into some unusual demand in your writing, then / will go a long way toward meeting your needs.

If you are a long-time LATEX/ user and have lots of mathematics in what you write, then you may recognize solutions for some familiar problems in this list of / features:

A convenient way to define new ioperator namej commands analogous to sin and lim, including proper side spacing and automatic selection of the correct font style and size (even when used in sub- or superscripts).

Multiple substitutes for the eqnarray environment to make various kinds of equation arrangements easier to write.

Equation numbers automatically adjust up or down to avoid overprinting on the equation contents (unlike eqnarray).

Spacing around equals signs matches the normal spacing in the equation environment (unlike eqnarray).

A way to produce multiline subscripts as are often used with summation or product symbols.

An easy way to substitute a variant equation number for a given equation instead of the automatically supplied number.

An easy way to produce subordinate equation numbers of the form (1.3a) (1. 3b) (1.3c) for selected groups of equations.

A boldsymbol command for printing bold versions of individual symbols, including things like * and lowercase Greek letters.

An amsthm package that provides a useful proof environment and some enhancements to the newtheorem command: support for multiple theorem styles in a single document and for unnumbered theorem types.

Chapter 1

How to use /

1 Using an AMS package in a LATEX/ document

A ipackagejpackage in LATEX/ terminology is an extension written in such a form that it can be used via the usepackage command. Many of the principal features of / are provided in separate packages so that they can be used individually on demand. The amsmath package is perhaps the single most noteworthy package, as it subsumes the amstext, amsbsy, and amsopn packages, and provides a number of other enhancements for mathematical typesetting. The current list of packages is:

amsmath Defines extra environments for multiline displayed equations, as well as a number of other enhancements for math.

amstext Provides a text command for typesetting a fragment of text inside a display.

amsbsy Defines boldsymbol and pmb ipoor manjs boldj commands.

amsopn Provides DeclareMathOperator for defining new ioperator namesj like sin and lim.

amsthm Provides a proof environment and extensions for the newtheorem command.

amsintx Provides more descriptive command syntax for integrals and sums.

amscd Provides a CD environment for simple commutative diagrams (no support for diagonal arrows).

amsxtra Provides certain odds and ends such as fracwithdelims and accentedsymbol.

upref Makes ref print cross-reference numbers always in an upright/roman font regardless of context.

2 Options for the amsmath package

The amsmath package has the following options:

centertags (default) For a split equation, place equation numbers vertically centered on the total height of the equation.

tbtags iTop-or-bottom tagsj: For a split equation, place equation numbers level with the last (resp. first) line, if numbers are on the right (resp. left).

sumlimits (default) Place the subscripts and superscripts of summation symbols above and below, in displayed equations. This option also affects other symbols of the same typen, *, *, *, and so forthnbut excluding integrals (see below).

nosumlimits Always place the subscripts and superscripts of summation-type symbols to the side, even in displayed equations.

intlimits Like sumlimits, but for integral symbols.

nointlimits (default) Opposite of intlimits.

namelimits (default) Like sumlimits, but for certain ioperator namesj such as det, inf, , max, min, that traditionally have subscripts placed underneath when they occur in a displayed equation.

nonamelimits Opposite of namelimits.

To use one of these package options, put the option name in the optional argument of the usepackage commandne.g., `\usepackage[intlimits]amsmath`.

The amsmath package also recognizes the following options which are normally selected (implicitly or explicitly) through the documentclass command, and thus need not be repeated in the option list of the usepackagenamsmathn statement.

leqno Place equation numbers on the left.

reqno Place equation numbers on the right.

fleqn Position equations at a fixed indent from the left margin rather than centered in the text column.

For symmetry there should perhaps be a `centereqn` option as well, to balance with `fleqn`, but as things currently stand there doesnjt seem to be a genuine need for it.

Chapter 2

Displayed equations (amsmath package)

1 Introduction

The amsmath package provides a number of additional displayed equation structures beyond the basic equation and eqnarray environments provided in basic LaTeX/. The augmented set includes:

```
  equation    align
  gather flalign
  multline    alignat
  split
```

(Although the standard eqnarray environment remains available, align or split are recommended instead.)

Except for split, each environment has both starred and unstarred forms, where the unstarred forms have automatic numbering using LaTeX/js equation counter. You can suppress the number on any particular line by putting notag before the
; you can also override it with a tag of your own using tagnnlabel>nn, where label> means arbitrary text such as n*n or niin used to knumberl the equation. There is also a tag* command that causes the text you supply to be typeset literally, without adding parentheses around it. tag and tag* can also be used within the unnumbered versions of all the amsmath alignment structures. Some examples of the use of tag may be found in the / sample files testmath.tex and subeqn.tex.

Table 1: Comparison of displayed equation environments (vertical lines indicating nominal margins)

```
\beginequation*
a=b
\endequation*
```

width.2pt 3pt 0pt $a=b$

height0pt width width.2pt

```
\beginequation
a=b
\endequation
```

width.2pt 3pt 0pt $a=b$ (1)

height0pt width width.2pt

```
\beginequation\labelxx
\beginsplit
a& =b+c-d\\
 & \quad +e-f\\
 & =g+h\\
 & =i
\endsplit
\endequation
```

width.2pt 3pt 0pt

[Sorry. Ignored `\beginsplit ... \endsplit`]

2)(

height0pt width width.2pt

```
\beginmultline
a+b+c+d+e+f\\
+i+j+k+l+m+n
\endmultline
```

width.2pt 3pt 0pt[Sorry. Ignored `\beginmultline ... \endmultline`]

height0pt width width.2pt

```
\begingather
a_1=b_1+c_1\\
a_2=b_2+c_2-d_2+e_2
\endgather
```

height0pt width width.2pt

```
\beginalign
a_1& =b_1+c_1\\
a_2& =b_2+c_2-d_2+e_2
\endalign
```

width.2pt 3pt 0pt (3)

(4)

height0pt width width.2pt

```
\beginalign
a_11& =b_11&
  a_12& =b_12\\
a_21& =b_21&
  a_22& =b_22+c_22
\endalign
```

width.2pt 3pt 0pt (5)

(6)

height0pt width width.2pt

```
\beginflalign*
a_11& =b_11&
  a_12& =b_12\\
a_21& =b_21&
  a_22& =b_22+c_22
\endflalign*
```

width.2pt 3pt 0pt[Sorry. Ignored \beginflalign* ... \endflalign*]
height0pt width width.2pt

2 Single equations

The equation environment is for a single equation with an automatically generated number. The equation* environment is the same except for omitting the number.

Basic LATEX doesnjt provide an equation* environment, but it has a functionally equivalent environment named displaymath.

3 Split equations without alignment

The multline environment is a variation of the equation environment used for equations that donjt fit on a single line. The first line of a multline will be at the left margin and the last line at the right margin, except for an indention on both sides in the amount of multlinegap. Intermediate lines will be centered independently within the display width. However, itjs possible to force a line to the left or right with commands shoveleft, shoveright. These commands take the entire line as an argument, up to but not including the final
; for example

[Sorry. Ignored \beginmultline ... \endmultline]

```
\beginmultline
\framebox[.65\columnwidth]A\\
\framebox[.5\columnwidth]B\\
\shoveright\framebox[.55\columnwidth]C\\
\framebox[.65\columnwidth]D
\endmultline
```

The value of multlinegap can be changed using LATEX/js setlength and addtolength commands. If the multline contains more than two lines, any lines other than the first and last will be centered individually between the margins (except when the fleqn option is in effect).

4 Split equations with alignment

Like multline, the split environment is for *single* equations that are too long to fit on one line and hence must be split into multiple lines. Unlike multline, however, the split environment provides for alignment among the split lines, using to mark alignment points, as usual. In addition, unlike the other amsmath equation structures, the split environment provides no numbering, because it is intended to be used only inside some other displayed equation structure, usually an equation, align, or gather environment, which provides the numbering. For example:

[Sorry. Ignored `\beginsplit ... \endsplit`]
(7)

```
\beginequation\labele:barwq\beginsplit
H_c&=\frac12n \sum^n_l=0(-1)^l(n-l)^p-2
\sum_l _1+\dots+ l _p=l\prod^p_i=1 \binomn_il _i\\
&\quad\cdot[(n-l )-(n_i-l _i)]^n_i-l _i\cdot
\Bigl[(n-l )^2-\sum^p_j=1(n_i-l _i)^2\Bigr].
\endsplit\endequation
```

5 Equation groups without alignment

The gather environment is used for a group of consecutive equations when there is no alignment desired among them; each one is centered separately within the text width (see Table ?).

6 Equation groups with mutual alignment

The align environment is used for two or more equations when vertical alignment is desired; usually binary relations such as equal signs are aligned (see Table ?).
To have several equation columns side-by-side, use extra ampersands to separate the columns:

x $=y$ X $=$
Y a $=$
$b+c$ (8

x' $=y'$ X' $=$
Y' a' $=$
b (9

$x+x'$ $=y+y'$ $X+X'$ $=$
$Y+Y'$ $a'b$ $=$
$c'b$ (1

```
\beginalign
x&=y  & X&=Y  & a&=b+c\\
x'&=y'& X'&=Y'& a'&=b\\
x+x'&=y+y' & X+X'&=Y+Y' & a'b&=c'b
\endalign
```

Line-by-line annotations on an equation can be done by judicious application of text inside an align environment:

?
) (11)

?
) (12)

(13)

```
\beginalign
x& = y_1-y_2+y_3-y_5+y_8-\dots
&& \textby \eqrefeq:C\\
 & = y'\circ y^*    && \textby \eqrefeq:D\\
 & = y(0) y'   && \text by Axiom 1.
\endalign
```

A variant environment alignat allows the space between equation columns to be explicitly specified. Here the number of equation columns must also be specified (where the number of icolumnsj is calculated as with number of & markers on any line).

[Sorry. Ignored `\beginalignat ... \endalignat`]

```
\beginalignat2
x& = y_1-y_2+y_3-y_5+y_8-\dots
   &\quad& \textby \eqrefeq:C\\
 & = y'\circ y^*  && \textby \eqrefeq:D\\
 & = y(0) y' && \text by Axiom 1.
\endalignat
```

7 Alignment building blocks

Some other equation alignment environments, such as aligned and gathered, construct self-contained units that can be used inside of other expressions, or set side-by-side. These environments take an optional argument to specify their vertical positioning with respect to the material on either side. The default is imiddlej placement with the vertical midpoint of the total unit falling on the math axis

The height of the cross-bar in the + symbol. For example:

[Sorry. Ignored `\beginaligned ... \endaligned`]

versus

[Sorry. Ignored `\beginaligned ... \endaligned`]

```
\beginequation*
\beginaligned
\alpha&=\alpha\alpha\\
\beta&=\beta\beta\beta\beta\beta\\
\gamma&=\gamma
\endaligned
\qquad\textversus\qquad
\beginaligned[t]
\delta&=\delta\delta\\
\eta&=\eta\eta\eta\eta\eta\eta\\
\varphi&=\varphi
\endaligned
\endequation*
```

kCasesl constructions like the following are common in mathematics:

[Sorry. Ignored `\begincases ... \endcases`]

(14)

and in the amsmath package there is a cases environment to make them easy to write:

```
P_r-j=\begincases
    0&  \textif $r-j$ is odd,\\
    r!\,(-1)^(r-j)/2&  \textif $r-j$ is even.
  \endcases
```

Notice the use of text and the embedded math.

8 Adjusting tag placement

Placing equation numbers can be a rather complex problem in multiline displays. The environments of the amsmath package try hard to avoid overprinting an equation number on the equation contents, if necessary moving the number down or up to a separate line. Even so, difficulties in accurately calculating the profile of an equation can occasionally result in a number placement that doesnjt look right. So there is a raisetag command provided to adjust the vertical position of the current equation number. To move a particular number up by six points, write n6ptn. (This kind of adjustment is fine tuning like line breaks and page breaks, and should therefore be left undone until your document is nearly finalized, or you may end up redoing the fine tuning several times to keep up with changing document contents.)

9 Vertical spacing and page breaks in multiline displays

You can use the
n[ndimension>n]n command to get extra vertical space between lines in all the amsmath displayed equation environments, as is usual in LATEX/. Unlike eqnarray, the amsmath environments donjt allow page breaks between lines, unless displaybreak or allowdisplaybreaks is used. The philosophy is that page breaks in such situations should receive individual attention from the author. displaybreak is best placed immediately before the
where it is to take effect. Like LATEX/js pagebreak, displaybreak takes an optional argument between 0 and 4 denoting the desirability of the pagebreak. n[0]n means kit is permissible to break herel without encouraging a break; displaybreak with no optional argument is the same as n[4]n and forces a break.

If you prefer a strategy of letting page breaks fall where they may, even in the middle of a multi-line equation, then you might put allowdisplaybreaks in the preamble of your document. An optional argument 1m4 can be used for finer control: n[1]n means allow page breaks, but avoid them as much as possible; values of 2,3,4 mean increasing permissiveness. When display breaks are enabled with allowdisplaybreaks, the

command can be used to prohibit a pagebreak after a given line, as usual.

10 Textual interjections within a display

The command intertext is used for a short interjection of one or two lines of text in the middle of a display alignment. Its salient feature is preservation of the alignment, which would not happen if you simply ended the display and then started it up again afterwards. intertext may only appear right after a

or

command. Notice the position of the word kandl in this example.

(15)

(16)

(17)

```
\beginalign
A_1&=N_0(\lambda;\Omega')-\phi(\lambda;\Omega'),\\
A_2&=\phi(\lambda;\Omega')-\phi(\lambda;\Omega),\\
\intertextand
A_3&=\mathcalN(\lambda;\omega).
\endalign
```

11 Equation numbering

11.1 Numbering hierarchy

In LaTeX/ if you wanted to have equations numbered within sectionsnthat is, have equation numbers (1.1), (1.2), ?, (2.1), (2.2), ?, in sections 1, 2, and so forthnyou could redefine theequation as suggested in the LaTeX/ manual [lm, §6.3, §C.8.4]:

```
\renewcommand\theequation\thesection.\arabicequation
```

This works pretty well, except that the equation counter wonjt be reset to zero at the beginning of a new section or chapter, unless you do it yourself using setcounter. To make this a little more convenient, the amsmath package provides a command numberwithin. To have equation numbering tied to section numbering, with automatic reset of the equation counter, the command would be

```
\numberwithinequationsection
```

11.2 Cross references to equation numbers

To make cross-references to equations easier, an eqref command is provided. This automatically supplies the parentheses around the equation number, and adds an italic correction if necessary. To refer to an equation that was labeled with the label ne: basetn, the usage would be n(?)n.

11.3 Subordinate numbering sequences

The amsmath package provides also a subequations environment to make it easy to number equations in a particular group with a subordinate numbering scheme. For example

```
\beginsubequations
...
\endsubequations
```

causes all numbered equations within that part of the document to be numbered (4.9a) (4.9b) (4.9c) ?, if the preceding numbered equation was (4.8). A label command immediately after `\beginsubequations` will produce a ref of the parent number 4.9, not 4.9a. The counters used by the subequations environment are `parentequation` and `equation` and addtocounter, setcounter, value, etc., can be applied as usual to those counter names. To get anything other than lowercase letters for the subordinate numbers, use standard LATEX/ methods for changing numbering style [lm, §6.3, §C.8.4]. For example, redefining theequation as follows will produce roman numerals.

```
\beginsubequations
\renewcommand\theequation\theparentequation \
romanequation
...
```

Chapter 3

Miscellaneous mathematics features (amsmath package)

1 Matrices

The amsmath package provides some environments for matrices beyond the basic array environment of LATEX/. The pmatrix, bmatrix, vmatrix and Vmatrix have (respectively) ( ), [ ], , and delimiters built in. For naming consistency there is a matrix environment sans delimiters. This is not entirely redundant with the array environment; the matrix environments all use more economical horizontal spacing than the rather prodigal spacing of the array environment. Also, unlike the array environment, you donjt have to give column specifications for any of the matrix environments; by default you can have up to 10 columns. More precisely: The maximum number of columns in a matrix is determined by the counter nMaxMatrixColsn (normal value = 10), which you can change if necessary using LATEX/js setcounter or addtocounter commands.(If you need left or right alignment in a column or other special formats you must resort to array.)

To produce a small matrix suitable for use in text, there is a smallmatrix environment (e.g., (

[Sorry. Ignored \beginsmallmatrix ... \endsmallmatrix]

)) that comes closer to fitting within a single text line than a normal matrix. Delimiters must be provided; there are no npn,nbn,nvn,nVn versions of smallmatrix. The above example was produced by

```
\bigl( \beginsmallmatrix
  a&b\\ c&d
\endsmallmatrix \bigr)
```

hdotsfornnnumber>nn produces a row of dots in a matrix spanning the given number of columns. For example,

[Sorry. Ignored \beginmatrix ... \endmatrix]

```
\beginmatrix a&b&c&d\\
e&\hdotsfor3 \endmatrix
```

The spacing of the dots can be varied through use of a square-bracket option, for example, n[1.5]3n. The number in square brackets will be used as a multiplier (i.e., the normal value is 1.0).

[Sorry. Ignored \beginpmatrix ... \endpmatrix]

, (1)

```
\beginpmatrix D_1t&-a_12t_2&\dots&-a_1nt_n\\
-a_21t_1&D_2t&\dots&-a_2nt_n\\
\hdotsfor[2]4\\
-a_n1t_1&-a_n2t_2&\dots&D_nt\endpmatrix
```

2 Math spacing commands

The amsmath package slightly extends the set of math spacing commands, as shown below. Both the spelled-out and abbreviated forms of these commands are robust, and they can also be used outside of math

[Sorry. Ignored \beginctab ... \endctab]

For the greatest possible control over math spacing, use mspace and imath unitsj. One math unit, or nmun, is equal to 1/18 em. Thus to get a negative quad you could write n-18.0mun.

3 Over and under arrows

Basic LATEX/ provides overrightarrow and overleftarrow commands. Some additional over and under arrow commands are provided by the amsmath package to fill out the set:

overleftarrow	underleftarrow
overrightarrow	underrightarrow
overleftrightarrow	underleftrightarrow

4 Dots

When the amsmath package is used, ellipsis dots should normally be typed as dots. Placement (on the baseline or centered) is determined by whatever follows the dots. If the next thing is a plus sign or other binary symbol, the dots will be centered; if itjs any other kind of symbol, they will be on the baseline.

If the dots fall at the end of a math formula, the next thing is something like end or nn or n|,*whichdoesnotgiveanyinformationabouthowtoplacethedots. Thenyoumusthelpbyusingdotscfor*k*dotswithcommas,"ordotsbfor*k*dotswithbinaryoperators/relations,"ordotsmfor*k*multiplicationdots,"ordotsifor*k*dotswithintegrals.*
"*Forexample,theinput*