mismath
Miscellaneous mathematical macros*

Antoine Missier
`antoine.missier@ac-toulouse.fr`

December 27, 2019

Contents

1 Introduction

According to the International Standards ISO 31-0:1992 to ISO 31-13:1992, superseded by ISO 80000-2:2009, mathematical constants e, i, π sould be typeset in upright shape and not in italic (sloping shape) like variables (see [1] [2] [3] [4]). This package provides some tools to achieve this (automatically).

Even if it is recommended to typeset vectors names in bold italic style [2] [3], they are often represented with arrows (particularly in school documents or in physics). To draw pretty arrows above vectors, we use the esvect package by Eddie Saudrais [7] and we provide a few more macros related to vectors with arrows, in particular to improve the typesetting of the norm: $\left\|\overrightarrow{AB}\right\|$ instead of LaTeX version $\|\overrightarrow{AB}\|$ which is not vertically adjusted, or worse $\left\|\overrightarrow{AB}\right\|$.

---

*This document corresponds to mismath v1.7, dated 2019/12/27. Thanks to François Bastouil for help in English translation.

The package also provides other macros for:

- some standard operator names,

- a few useful aliases,

- improving some spacings in mathematical formulas,

- systems of equations and small matrices,

- displaymath in double columns for long calculation.

To avoid incompatibility, a large majority of our macros will be defined only if there is not another command with the same name in the packages loaded before mismath. If a macro is already defined, compilation will produce a warning message and mismath definition will simply be ignored. To keep mismath command, either load mismath before the other package with which it is in conflict for the name of the command (assuming the other package supports it), or use `\let\`⟨*command*⟩`\relax` before loading mismath.

⟨*options*⟩   The amsmath package is loaded by mismath without option. For using amsmath with options (see [5]), these options can be added when calling mismath, or amsmath has to be loaded with the required options before mismath.

Another package, mathtools by Morten Høgholm and Lars Madsen [6] is also loaded. It provides many usefull macros.

A recommendation, seldom observed, is to typeset uppercase Greek letters in italic shape like other variables [3]. This is automatically done with the fixmath package by Walter Schmidt [9], but this feature is not implemented in mismath because this rule is conflicting to the one used for instance in France where all mathematics capitals have to be typeset in upright shape[1]. The choice of loading or not one of these packages remains thus to the user.

2   Usage

2.1   Mathematical constants

\mathup   As for classic functions identifiers, *predefined* mathematical constants should be typeset in upright shape (generally in roman family), even if this practice is not really common and tedious to respect. To avoid to stuff a document with

\e   `\mathrm{e}` or `\mathrm{i}` (or better `\mathup{e}` and `\mathup{i}`[2]), the pack-

\i   age provides `\e` command for the base of the natural logarithm and `\i` or `\j` for

\j   imaginary numbers. One can notice that `\i` and `\j` already exist in LaTeX: using in LR mode, they produce "ı" and "ȷ" without the point so one can place accents

---

[1]The frenchmath package [14] takes this rule into account.

[2]`\mathup` is based on `\operatorfont` (from amsopn package, automatically loaded by amsmath). The beamer package uses a default sans serif math font, but `\mathrm` produces a font with serif in beamer. This problem is solved by using `\mathup` instead of `\mathrm`.

on them, and in mathematical mode they produce "Latex warning: Command invalid in math mode". Redefining \i and \j only concerns mathematical mode.

\enumber
\inumber
\jnumber

Nevertheless, it can be tiresome to type a lot of backslashes in a document with many formulas containing e or i. So a way is proposed here to free of it by placing \enumber, \inumber or \jnumber in the preamble: e, i or j will then automatically be set in upright shape in the whole document, no need to type \e, \i or \j, let's hope that there are not many other *e*, *i* or *j* as variables. However, one can still get italicized *e*, *i* or *j* with LaTeX command \mathit or \mathnormal. Of course, this does not fully comply with LaTeX philosophy: in the document body, objects should be pointed out by their nature rather than their typographical characteristics, defined in the preamble. But these macros are really handy and thanks to them it is possible to bring a document up to the standards afterwards; besides anyone is free to use them or not.

\pinumber[⟨*font*⟩]

Mathematical constant π should also be typeset in upright shape (see [3] and [4]), which differs from italicized π. This recommendation is even less observed than the one concerning e and i [1]. The upgreek package by Walter Schmidt [8] makes it possible to typeset greek letters in upright font by using commands such as \upalpha, \upbeta,... To avoid typing a lot of \uppi, we provide the \pinumber macro, which has to be put in the preamble. This command loads the upgreek package with an optional ⟨*font*⟩ argument: Symbol (by default), Euler or Symbolsmallscale (see [8]). It also redefines the \pi command to typeset all \pi in the selected upright font.

By activating \enumer, \inumber and \pinumber in the preamble, you can get for instance:

$$\texttt{\$e\^{}\{i\textbackslash pi\} = -1\$} \qquad\qquad e^{i\pi} = -1$$

\itpi

When \pinumber is activated, the original italic π is still available with \itpi.

2.2   Vectors

\vect

By default, the \vect command[3], produces vectors with arrows (thanks to the esvect package by Eddie Saudrais[4]) which are much more elegant than those produced by LaTeX \overrightarrow command (giving $\overrightarrow{AB}$). The esvect package has an optional argument (one letter between a and h) defining the required type of arrow (see [7]). In mismath, esvect is loaded with the option b: \vect{AB} gives $\overrightarrow{AB}$. To choose another type of arrow, esvect must be called with the required option *before* mismath, for instance \usepackage[d]{esvect} will give the arrows produced by default in [7].

\boldvect

\vect makes also possible to typeset vector's names using bold italic (according to ISO recommendation [4]) rather than arrows. For this, calling \boldvect will modify the behavior of \vect:

---

[3] As for many macros of this package, the definition will take effect only if this macro is not defined before by another package.

[4] esvect provides \vv macro used by \vect.

```
\[ \boldvect \vect{v}
   =\lambda\vect{e}_x+\mu\vect{e}_y. \]
```
$$\boldsymbol{v} = \lambda\boldsymbol{e}_x + \mu\boldsymbol{e}_y.$$

`\boldvectcommand` By default `\boldvect` uses the `\boldsymbol` command[5] from amsbsy package, loaded by amsmath. But other packages producing bold italic can be preferred, e.g. `\bm` from bm package or `\mathbold` from fixmath package or `\mathbfit` from isomath. For that, redefine `\boldvectcommand`: for instance `\renewcommand\boldvectcommand{\mathbold}`.

By setting `\boldvectcommand` to `\mathbf`, `\vect` produces vectors in bold *upright* shape, which tends to be used instead of bold *italic* (but probably for bad reasons).

`\arrowvect` At any moment, you can get back to the default behavior with the inverse switch `\arrowvect`. These switches can be placed anywhere: inside mathematical mode or inside an environment (with local effect) or outside (with global effect).

`\hvect` When vectors with arrows are typeset side by side, arrows can be set up a bit higher (with a vertical phantom box containing h) to avoid inelegants effects:

- $\overrightarrow{AB} = \vec{u} + \overrightarrow{AC}$ is less than $\overrightarrow{AB} = \vec{u} + \overrightarrow{AC}$, obtained with `\hvect{u}`;

- $\vec{a} \cdot \vec{b} = 0$ is less than $\vec{a} \cdot \vec{b} = 0$, obtained with `\hvect{a}`.

The `\boldvect` switch has no effect on the `\hvect` macro which always typesets arrows on vectors (with the `\vv` command from the esvect package).

`\hvec` In a similar way, `\hvec` raises the little arrow produced by the LaTeX command `\vec` (but only from height of t letter):

- $\mathcal{P} = \vec{f} \cdot \vec{v}$ is less than $\mathcal{P} = \vec{f} \cdot \vec{v}$, obtained with `\hvec{v}`;

- $\vec{f} = m\vec{a}$ is less than $\vec{f} = m\vec{a}$, obtained with `\hvec{a}`.

`\norm` The norm of a vector is classically produced by the delimiters `\lVert` and `\rVert` (rather than \|) or `\left\Vert` and `\right\Vert` for delimiters adapting to the content. Unfortunately, these delimiters are always vertically centred, relatively to the middle of the base line, whereas vectors with arrows are asymetrics objects, the height above the middle of the base line being superior to the depth under it. The code `$\norm{\vec{h}}$` raises the double bar to produce $\left\|\vec{h}\right\|$. Let's notice that the height of the bars don't adjust to content, but however to context: main text, subscripts or exponents.

2.3 Standard operator names

`\di` The *differential* operator should be typeset in upright shape and not in italic, to make it different from variables (as mentioned in [1] [2] [3] [15]). For this, we provide the `\di` command. See the following examples (notice the thin spaces before the d, as for classic function's names):

---

[5]`\mathbf` gives upright bold font, even if used in combination with `\mathit`.

4

```
\[ \iint xy\di x\di y \]
```

$$\iint xy \, \mathrm{d}x \, \mathrm{d}y$$

```
\[ m\frac{\di^2x}{\di t^2}
   + h\frac{\di x}{\di t} + kx = 0 \]
```

$$m\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + h\frac{\mathrm{d}x}{\mathrm{d}t} + kx = 0$$

This command can also stand for *distance* (hence its name):

$$\lambda \, \mathrm{d}(A, \mathcal{F}) + \mu \, \mathrm{d}(B, \mathcal{H}).$$

`\P`
`\E`
`\V`
To refer to probability[6] and expectation the proper use is to typeset capital letters P, E in upright shape as for any standard function identifier. This is obtained with `\P` and `\E`. Variance is normally denoted by Var (see further), but in some countries we can find V produced by `\V`.

`\Par`
The `\P` command already existed to refer to the end of paragraph symbol ¶ and has been redefined, but this symbol can still be obtained with `\Par`.

`\probastyle`
Some authors use "blackboard bold" font to represent probability, expectation and variance: $\mathbb{P}, \mathbb{E}, \mathbb{V}$. The `\probastyle` macro sets the appearance of `\P`, `\E` and `\V`: for instance `\renewcommand\probastyle{\mathbb}`[7] brings the previous "openwork" letters. `\mathbb` comes from amsfonts package (loaded by amssymb but also available standalone) which has to be called in the preamble.

The following operator names are also defined in mismath:

`\adj`	adj	`\erf`	erf	`\rank`	rank
`\Aut`	Aut	`\grad`	$\overrightarrow{\mathrm{grad}}$	`\Re`	$\overrightarrow{\mathrm{Re}}$
`\Conv`	Conv	`\id`	id	`\rot`	$\overrightarrow{\mathrm{rot}}$
`\cov`	cov	`\Id`	Id	`\sgn`	sgn
`\Cov`	Cov	`\im`	im	`\spa`	span
`\curl`	$\overrightarrow{\mathrm{curl}}$	`\Im`	Im	`\tr`	tr
`\divg`	div	`\lb`	lb	`\Var`	Var
`\End`	End	`\lcm`	lcm	`\Zu`	Z

By default, operators returning vectors, `\grad` and `\curl` (or its synonym `\rot` rather used in Europe), are written with an arrow on the top. When `\boldvect` is activated, they are typeset in bold style: **grad**, **curl**, **rot**. For the covariance and the identity function, two notations are proposed, with or without a first capital letter, because they are both very common. On the other hand, "im" stands for the image of a linear transformation (like "ker" for the kernel) but "Im" is the imaginary part of a complex number. Notice that `\div` and `\span` already exist and haven't been redefined, therefore the `\divg` and `\spa` macros; `\Z` is used otherwise (see further), therefore `\Zu`, to designate the center of a group: Z(G) (from German Zentrum).

`\oldRe`
`\oldIm`
The `\Re` and `\Im` macros already existed, to refer to real and imaginary part of a complex number, producing outdated symbols $\Re$ and $\Im$. They have been

---

[6]LATEX provides also `\Pr` which gives Pr.
[7]As for `\boldvect` and `\arrowvect`, effect is local to the container environment.

redefined according to actual use, as mentionned in the above table, but it's still possible to get the old symbols with `\oldRe` and `\oldIm`.

Some (inverse) circular or hyperbolic functions, missing in LaTeX, are also provided by mismath:

`\arccot`	arccot	`\arsinh`	arsinh	`\arcoth`	arcoth
`\sech`	sech	`\arcosh`	arcosh	`\arsech`	arsech
`\csch`	csch	`\artanh`	artanh	`\arcsch`	arcsch

`\bigO`
`\bigo`
`\lito`

Asymptotic comparison operators (in Landau notation) are obtained with `\bigO` or `\bigo` and `\lito` commands:

$$n^2 + \mathcal{O}(n \log n) \quad \text{or} \quad n^2 + \mathrm{O}(n \log n) \quad \text{and} \quad \mathrm{e}^x = 1 + x + \mathrm{o}(x^2).$$

2.4   A few useful aliases

In the tradition of Bourbaki and D. Knuth, proper use requires that classics sets of numbers are typeset in bold roman: $\mathbf{R}, \mathbf{C}, \mathbf{Z}, \mathbf{N}, \mathbf{Q}$, "openwork" letters ($\mathbb{R}, \mathbb{Z}, \ldots$) being restricted to writing at blakboard [15]; and likewise to designate a field: $\mathbf{F}$ or $\mathbf{K}$ (Körper in German). We get these symbols with the following macros:

$$\text{\textbackslash R, \textbackslash C, \textbackslash Z, \textbackslash N, \textbackslash Q, \textbackslash F, \textbackslash K.}$$

`\mathset`

The `\mathset` command enables to change in a global way the behavior of all these macros: by default, `\mathset` is an alias for `\mathbf`, but if one prefer openwork letters, just place `\renewcommand\mathset{\mathbb}` in the preamble, after loading amsfonts package (which provides the "blackboard bold" typeface, also loaded by amssymb).

`\ds`

The `\displaystyle` command being very common, alias `\ds` is provided. Not only it eases typing but also it makes source code more readable.

Symbols with limits behave differently for in-line formulas or for displayed equations. In the latter case, "limits" are put under or above whereas for in-line math mode, they are placed on the right, as subscript or exponent. Compare: $\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$ with

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}.$$

`\dlim`
`\dsum`
`\dprod`
`\dcup`
`\dcap`

With in-line math mode, displaymath behavior can be forced with `\displaystyle` or its alias `\ds`, but then, all the rest of the current mathematical environment will be set in displaymath mode too (in the previous example, the fraction will be expanded). Just like the amsmath command `\dfrac` only transforms the required fraction in display style, we can limit display style effect to the affected symbol, by using the following macros: `\dlim`, `\dsum`, `\dprod`, `\dcup`, `\dcap`. So `$\dlim_{x\to +\infty}\frac{1}{x}$` gives $\lim\limits_{x \to +\infty} \frac{1}{x}$.

`\lbar`
`\hlbar`

Large bars over expressions are obtained with `\overline` or, shorter, its alias `\lbar`, to get for instance $\overline{z_1 z_2}$. Such as for vectors, one can raise the bar (from the height of h) with the `\hlbar` command, in order to correct uneven bars heights.

$\overline{z + z'} = \overline{z} + \overline{z'}$ is less than $\overline{z+z'} = \overline{z} + \overline{z'}$, obtained with `\hlbar{z}`.

\eqdef   The `\eqdef` macro writes equality symbol topped with "def" (thanks to the LaTeX command `\stackrel`):

```
$ \e^{\i\theta} \eqdef
  \cos\theta + \i\sin\theta $
```
$$\mathrm{e}^{\mathrm{i}\theta} \stackrel{\mathrm{def}}{=} \cos\theta + \mathrm{i}\sin\theta$$

\unbr   `\unbr` is an alias for `\underbrace`[8], making source code more compact.

```
$ (QAP)^n = \unbr{QAP\mul QAP\mul
  \cdots\mul QAP}_{n\text{ times}} $
```
$$(QAP)^n = \underbrace{QAP\times QAP\times \cdots \times QAP}_{n \text{ times}}$$

\iif   `\iif` is an alias for "if and only if", to be used in text mode.

2.5   Improving some spacings in mathematical formulas

\mul   The multiplication symbol obtained with `\times` produces the same spacing than addition or substraction operators, whereas division obtained with / is closer to its arguments. This actually hides the priority of the multiplication on $+$ and $-$. This is why we provide the `\mul` macro, behaving like / (ordinary symbol) and leaving less space around than `\times`:

$$\lambda + \alpha \times b - \beta \times c \text{ is less than } \lambda + \alpha{\times}b - \beta{\times}c, \text{ obtained with } \texttt{\textbackslash mul}.$$

When using `\mul` before an operator name or a `\left...\right` structure, additionnal spacing occur on the right side of `\mul`. A solution to get the same amount of space on the two sides of `\mul`, is to enclose the operator name (or the structure) with brackets:

$$\text{Compare } x\times \sin x \text{ with } x{\times}\sin x \text{ obtained with } \texttt{x\textbackslash mul\{\textbackslash sin x\}}.$$

\then   The `\then` macro produces the symbol $\implies$ surrounded by large spaces as the
\txt   standard macro `\iff` does it with $\iff$. In a similar way, `\txt` based on the `\text` macro (from the amstext package, automatically loaded by amsmath), leaves em quad spaces (`\quad`) around the text. See the following example:

```
\ln x=a\then x=\e^a \txt{rather than} \ln x=a\Longrightarrow x=\e^a
```
$$\ln x = a \implies x = \mathrm{e}^a \quad \text{rather than} \quad \ln x = a \Longrightarrow x = \mathrm{e}^a$$

\paren   Spaces around parenthesis produced by `\left(...\right)` may be too large, for example after a function name or a point name with coordinates. A solution is to add a thin negative space `\!` before the opening (or after the closing) parenthesis, or to enclose the `\left(...\right)` structure by brackets, or to use the `\paren` macro:

$$\sin\left(\tfrac{\pi}{3}\right)\times 2 \text{ is less than } \sin\!\left(\tfrac{\pi}{3}\right)\!\times 2 \text{ obtained with}$$
```
\sin\paren{\frac{\pi}{3}}\mul 2.
```

---

[8] The mathtools package by Morten Høgholm and Lars Madsen [6] provides a new improved version of `\underbrace` command (as many other usefull macros); it is loaded by mismath.

\pow   When typesetting an exponent after a closing *big* parenthesis produced by `\right)`, the exponent is little to far from the parenthesis. The command `\pow{⟨expr⟩}{⟨pow⟩}` sets ⟨expr⟩ between parentheses and puts the exponent ⟨pow⟩ slightly closer to the right parenthesis[9]. Compare:

$$\mathrm{e}^a \sim \left(1 + \frac{a}{n}\right)^n \quad \text{and} \quad \mathrm{e}^a \sim \left(1 + \frac{a}{n}\right)^{\!n}.$$

\abs   Absolute value (or modular for a complex number) should be typeset with `\lvert ... \rvert` rather than | which doesn't respect correct spaces for delimiters; for bars whose height has to adapt to content, we use `\left\vert ... \right\vert` or, more simply, the `\abs{...}` command which is equivalent[10].

\lfrac   This macro behaves like `\frac` but with medium spaces around the arguments, so the corresponding fraction bar is perceptibly a little bit longer:

```
\[ \lbar{Z} =
    \lfrac{\lbar{z_1-z_2}}{\lbar{z_1+z_2}} \]
```

$$\overline{Z} = \frac{\overline{z_1 - z_2}}{\overline{z_1 + z_2}}$$

[ ]   Brackets symbols [and] have been redefined for mathematical mode because, in standard LATEX, the space before them can be unsuitable[11]:

```
$x\in ]0,\pi[ \cup ]2\pi,3\pi[$
```

$$\begin{cases} x \in]0, \pi[\cup]2\pi, 3\pi[& \text{without mismath} \\ x \in]0, \pi[\cup]2\pi, 3\pi[& \text{with mismath} \end{cases}$$

In our code, [and] symbols are not defined anymore as delimiters. One can regret it because a line break could occur between the two, but in addition to the fact that it works very well like that for spaces (because these symbols are most of the time preceded or followed by relational, binary or punctuation symbols), it is always possible to transform them into delimiters with `\left` and `\right`[12].

2.6 Environments for systems of equations and small matrices

system   The `system` environment produces a system of equations:

```
$\begin{system}
    x=1+2t \\ y=2-t \\ z=-3-t
\end{system}$
```

$$\begin{cases} x = 1 + 2t \\ y = 2 - t \\ z = -3 - t \end{cases}$$

\systemsep   This first example could also have been produced with `cases` environment from `amsmath` package, although `cases` places mathematical expressions closer to the bracket (which makes sense considering it's use). `\systemsep` enables to set the

---

[9]This macro gives bad results with normal sized parenthesis.

[10]Another solution is to define `\abs` with the `\DeclarePairedDelimiter` command from the `mathtool` package [6].

[11]The interval package [11] gives another solution, less direct, based on an `\interval` macro.

[12]Is LATEX definition of [as `mathopen` really appropriate where this symbol could almost also logically have been defined as `mathclose`?

gap between the bracket and the expressions, set by default to `\medspace`. This gap may be reduce, for instance: `\renewcommand{\systemsep}{\thinspace}`, or enlarged with `\thickspace` (and with `\renewcommand\systemsep}{}` we get back to what `cases` do).

`system[⟨coldef⟩]`   By default, a system is written like an `array` environment with only one column, left aligned. The environment has an optional argument to create several columns, specifying their alignment, with the same syntax than the `array` environment of LaTeX: `\begin{system}[cl]` produces a two-column system, the first one being centred, the second being left aligned, such as in the following example:

```
$\begin{system}[cl]
    y & =\dfrac{1}{2}x-2 \\[1ex]
    (x,y) & \neq (0,-2)
\end{system}$
```

$$\begin{cases} y & = \dfrac{1}{2}x - 2 \\ (x,y) \neq (0,-2) \end{cases}$$

`\systemstretch`   Default spacing between the lines of a `system` environment has been slightly enlarged compared to the one from `array` environments (from 1.2 factor). This spacing may be changed by typing `\renewcommand{\systemstretch}{⟨stretch⟩}`, inside the current mathematical environment (for a local change) or outside (for a global change). By default, stretch's value is 1.2. In addition we can use a carriage return with a spacing option such as it has been done above with `\\[1ex]`.

Another example with `\begin{system}[rl@{\quad}l]`[13]:

$$\begin{cases} x + 3y + 5z = 0 & R_1 \\ 2x + 2y - z = 3 & R_2 \\ 3x - y + z = 2 & R_3 \end{cases} \iff \begin{cases} x + 3y + 5z = 0 & R_1 \\ 4y + 11z = 3 & R_2 \leftarrow 2R_1 - R_2 \\ 5y + 7z = -1 & R_3 \leftarrow \frac{1}{2}\left(3R_1 - R_3\right) \end{cases}$$

Let's mention the **systeme** package [12] which deals with linear systems with a lighter syntax and automatic alignments on $+$, $-$, $=$, and also the **spalign** package [13] which moreover produces nice alignments for matrices (with spaces and semicolons as delimiters).

`spmatrix`   The **amsmath** package provides various environments to typeset matrices: for instance `pmatrix` surrounds the matrix with parenthesis or `smallmatrix` typesets a small matrix that can even be inserted in a text line. We provide a combination of the two with `spmatrix`:

`$\vec{u}\begin{spmatrix}-1\\2\end{spmatrix}$` yielding $\vec{u}\left(\begin{smallmatrix} -1 \\ 2 \end{smallmatrix}\right)$.

The **mathtools** package enhance **amsmath** matrices environments and provides also a small matrix environment with parenthesis. Furthermore, with starred version `\begin{psmallmatrix*}[⟨col⟩]`, you can choose the alignment inside the columns (`c`, `l` or `r`). But sadly, the space before the left parenthesis is too narrow regarding to the space inside the parenthesis. Compare previous $\vec{u}\left(\begin{smallmatrix} -1 \\ 2 \end{smallmatrix}\right)$ with $\vec{u}(\begin{smallmatrix} -1 \\ 2 \end{smallmatrix})$.

---

[13] `@{...}` sets inter-column space.

2.7 Displaymath in double columns

mathcols The `mathcols` environment activates mathematical mode and enables to arrange "long" calculation in double columns, separated with a central rule, as shown in the following example. But you have to load the multicol package in the preamble.

$$\frac{1}{2\times\left(\frac{1}{4}\right)^n + 1} \geq 0.999$$
$$\iff 1 \geq 1.998\left(\frac{1}{4}\right)^n + 0.999$$
$$\iff 0.001 \geq \frac{1.998}{4^n}$$

$$\iff 4^n \geq 1998$$
$$\iff n\ln 4 \geq \ln(1998)$$
$$\iff n \geq \frac{\ln(1998)}{\ln 4} \approx 5.4$$
$$\iff n \geq 6$$

\changecol   The `\changecol` macro causes a change of column; aligment is produced using the classic delimiters `&` and `\\`.

```
\begin{mathcols}
        & \frac{1}{2 \mul {\pow{\frac{1}{4}}{n}} + 1} \geq 0.999 \\
    \iff\ & 1 \geq 1.998  \pow{\frac{1}{4}}{n} + 0.999 \\
    \iff\ & 0.001 \geq \frac{1.998}{4^n} \\
\changecol
    & \iff 4^n \geq 1998 \\
    & \iff n \ln 4 \geq \ln(1998) \\
    & \iff n \geq \frac{\ln(1998)}{\ln 4} \approx 5.4 \\
    & \iff n \geq 6
\end{mathcols}
```

3   Implementation

```
1 \DeclareOption*{\PassOptionsToPackage{\CurrentOption}{amsmath}}
2 \ProcessOptions \relax
3 \@ifpackageloaded{amsmath}{}{\RequirePackage{amsmath}}
4 \@ifpackageloaded{esvect}{}{\RequirePackage[b]{esvect}}
5 \RequirePackage{ifthen}
6 \RequirePackage{xspace}
7 \RequirePackage{mathtools}
```

The above conditional packages loading avoids "option clash" errors if the packages have been previously loaded with (other) options.

\bslash   The three following internal commands are meta commands for a conditional macro definition with warning message if the macro already exists. The `\bslash` macro used inside `\@mwarning` comes from doc.sty package by Frank Mittelbach. It can also be used in other documents instead of `\textbackslash` (which doesn't work here).

```
8 {\catcode`\|=\z@ \catcode`\\=12 |gdef|bslash{\}} % the \bslash command
9 \newcommand\@mwarning[1]{
10      \PackageWarning{mismath}{
```

```
11          Command \bslash #1 already exist and will not be redefined
12      }
13 }
14 \newcommand\@mmacro[2]{
15      \@ifundefined{#1}{
16          \expandafter\def\csname #1\endcsname{#2}
17      }{\@mwarning{#1}}
18 }
19 \newcommand\@moperator[3][]{% this macro is ugly, TODO: by default #1=#3
20      \ifthenelse{\equal{#1}{}}{
21          \@ifundefined{#3}{
22              \DeclareMathOperator{#2}{#3}
23          }{\@mwarning{#3}}
24      }{
25          \@ifundefined{#1}{
26              \DeclareMathOperator{#2}{#3}
27          }{\@mwarning{#1}}
28      }
29 }
30
```

To work correctly with the beamer package, we did not use \mathrm but \mathup (based on \operatorfont from the mathopn package) to produce the correct upright shape font. This command works also fine with other sans serif fonts like cmbright. Moreover for beamer, \enumber must use the family default font defined by the beamer package (sans serif), therefore the \AtBeginDocument inside the macro (otherwise it has no effect). The same holds for \inumber and \jnumber.

\AtBeginDocument is also necessary to redefine \i when calling the hyperref package which overwrites the \i definition.

```
31 \providecommand{\mathup}[1]{{\operatorfont #1}}
32 \@mmacro{e}{\mathup{e}}
33 \AtBeginDocument{\let\oldi\i \let\oldj\j
34      \renewcommand{\i}{\TextOrMath{\oldi}{\mathup{i}}}
35      \renewcommand{\j}{\TextOrMath{\oldj}{\mathup{j}}}
36 }
37
38 \DeclareSymbolFont{UpSh}{\encodingdefault}{\familydefault}{m}{n}
39 \newcommand{\enumber}{
40      \AtBeginDocument{\DeclareMathSymbol{e}\mathalpha{UpSh}{`e}}
41 }
42 \newcommand{\inumber}{
43      \AtBeginDocument{\DeclareMathSymbol{i}\mathalpha{UpSh}{`i}}
44 }
45 \newcommand{\jnumber}{
46      \AtBeginDocument{\DeclareMathSymbol{j}\mathalpha{UpSh}{`j}}
47 }
48 \newcommand*{\pinumber}[1][Symbol]{
49      \@ifpackageloaded{upgreek}{}{\usepackage[#1]{upgreek}}
```

```
50     \let\itpi\pi
51     \renewcommand{\pi}{\uppi}
52 }
53
54 \newboolean{arrowvect}
55 \setboolean{arrowvect}{true}
56 \newcommand{\arrowvect}{\setboolean{arrowvect}{true}}
57 \newcommand{\boldvect}{\setboolean{arrowvect}{false}}
58 \newcommand{\boldvectcommand}{\boldsymbol} % needs bm package
59 \@mmacro{vect}{\ifthenelse{\boolean{arrowvect}}{\vv}{\boldvectcommand}}
60 \newcommand*{\hvect}[1]{\vv{\vphantom{h}#1}}
61 \newcommand*{\hvec}[1]{\vec{\vphantom{t}#1}}
62
63 \newcommand*{\@norm}[1]{
64     \mbox{\raisebox{1.75pt}{$\bigl\Vert$}} #1
65     \mbox{\raisebox{1.75pt}{$\bigr\Vert$}} }
66 % works better than with relative length
67 \newcommand*{\@@norm}[1]{
68     \mbox{\footnotesize\raisebox{1pt}{$\Vert$}} #1
69     \mbox{\footnotesize\raisebox{1pt}{$\Vert$}} }
70 \newcommand*{\@@@norm}[1]{
71     \mbox{\tiny\raisebox{1pt}{$\Vert$}} #1
72     \mbox{\tiny\raisebox{1pt}{$\Vert$}} }
73 \providecommand*{\norm}[1]{
74     \mathchoice{\@norm{#1}}{\@norm{#1}}{\@@norm{#1}}{\@@@norm{#1}} }
75
76 \newcommand{\di}{\mathop{}\!\mathup{d}}
77 \newcommand\probastyle{}
78 \let\Par\P % end of paragraph symbol
79 \renewcommand{\P}{\operatorname{\probastyle{P}}}
80 \@mmacro{E}{\operatorname{\probastyle{E}}}
81 \@mmacro{V}{\operatorname{\probastyle{V}}}
82 \newcommand{\PEupright}{
83     \AtBeginDocument{% necessary for working with beamer
84         \DeclareMathSymbol{P}\mathalpha{UpSh}{`P}
85         \DeclareMathSymbol{E}\mathalpha{UpSh}{`E}
86     }
87 }
88
89 \@moperator{\adj}{adj}
90 \@moperator{\Aut}{Aut}
91 \@moperator{\Conv}{Conv}
92 \@moperator{\cov}{cov}
93 \@moperator{\Cov}{Cov}
94 \@mmacro{curl}{\operatorname{\vect{\mathup{curl}}}}
95 \@moperator[divg]{\divg}{div}
96 \@moperator{\End}{End}
97
98 \@moperator{\erf}{erf}
99 \@mmacro{grad}{\operatorname{\vect{\mathup{grad}}}}
```

```latex
100 \@moperator{\id}{id} % mathop or mathord ?
101 \@moperator{\Id}{Id}
102 \@moperator{\im}{im}
103 \let\oldIm\Im \renewcommand{\Im}{\operatorname{Im}}
104 \@moperator{\lb}{lb}
105 \@moperator{\lcm}{lcm}
106
107 \@moperator{\rank}{rank}
108 \let\oldRe\Re \renewcommand{\Re}{\operatorname{Re}}
109 \@mmacro{rot}{\operatorname{\vect{\mathup{rot}}}}
110 \@moperator{\sgn}{sgn}
111 \@moperator[spa]{\spa}{span}
112 \@moperator{\tr}{tr}
113 \@moperator{\Var}{Var}
114 \@moperator[Zu]{\Zu}{Z}
115
116 \@moperator{\arccot}{arccot}
117 \@moperator{\sech}{sech}
118 \@moperator{\csch}{csch}
119 \@moperator{\arsinh}{arsinh}
120 \@moperator{\arcosh}{arcosh}
121 \@moperator{\artanh}{artanh}
122 \@moperator{\arcoth}{arcoth}
123 \@moperator{\arsech}{arsech}
124 \@moperator{\arcsch}{arcsch}
125
126 \@moperator[bigO]{\bigO}{\mathcal{O}}
127 \@moperator[bigo]{\bigo}{O}
128 \@moperator[lito]{\lito}{o}
129
130 \newcommand{\mathset}{\mathbf}
131 \@mmacro{R}{\ensuremath{\mathset{R}}\xspace}
132 \@mmacro{C}{\ensuremath{\mathset{C}}\xspace}
133 \@mmacro{N}{\ensuremath{\mathset{N}}\xspace}
134 \@mmacro{Z}{\ensuremath{\mathset{Z}}\xspace}
135 \@mmacro{Q}{\ensuremath{\mathset{Q}}\xspace}
136 \@mmacro{F}{\ensuremath{\mathset{F}}\xspace}
137 \@mmacro{K}{\ensuremath{\mathset{K}}\xspace}
138
139 \@mmacro{ds}{\displaystyle}
140 \@mmacro{dlim}{\lim\limits}
141 \@mmacro{dsum}{\sum\limits}
142 \@mmacro{dprod}{\prod\limits}
143 \@mmacro{dcup}{\bigcup\limits}
144 \@mmacro{dcap}{\bigcap\limits}
145 \@mmacro{lbar}{\overline}
146 \providecommand*{\hlbar}[1]{\overline{\vphantom{h}#1}}
147 \@mmacro{eqdef}{\stackrel{\mathup{def}}{=}}
148 \@mmacro{unbr}{\underbrace}
149 \@mmacro{iif}{if and only if\xspace}
```

```
150
151 \@mmacro{mul}{\mathord{\times}}
152 \@mmacro{then}{\ \Longrightarrow \ \mbox{} }
```

Without `\mbox{}`, the space produced by `\` would be suppressed in tables.

```
153 \providecommand*{\txt}[1]{\quad\text{#1}\quad}
154 \providecommand*{\paren}[1]{\mathopen{\left(#1\right)}}
155 \providecommand*{\pow}[2]{\left( #1 \right)^{\!#2}}
156 \providecommand*{\abs}[1]{\left\vert#1\right\vert}
157 \providecommand*{\lfrac}[2]{\frac{\:#1\:}{\:#2\:}}
158 \DeclareMathSymbol{]}\mathord{UpSh}{093} % originally \mathclose
159 \DeclareMathSymbol{[}\mathord{UpSh}{091} % originally \mathopen
160
161 \newcommand{\systemstretch}{1.2}
162 \newcommand{\systemsep}{\medspace}
163 \newenvironment{system}[1][l]{
164     \renewcommand{\arraystretch}{\systemstretch}
165     \setlength{\arraycolsep}{0.15em}
166     \left\{\begin{array}{@{\systemsep}#1@{}} %
167 }{\end{array}\right.}
168
169 \newenvironment{spmatrix}{
170     \left(\begin{smallmatrix}
171 }{\end{smallmatrix}\right)}
172
173 \newenvironment{mathcols}{% needs multicol package
174     \renewcommand{\columnseprule}{0.1pt}
175     \begin{multicols}{2}
176         \par\noindent\hfill
177         \begin{math}\begin{aligned}\displaystyle
178 }{%
179         \end{aligned}\end{math} \hfill\mbox{}
180     \end{multicols}
181 }
182 \newcommand{\changecol}{%
183     \end{aligned}\end{math} \hfill\mbox{}
184     \par\noindent\hfill
185     \begin{math}\begin{aligned}\displaystyle
186 }
```

References

[1] *Typesetting mathematics for science and technology according to ISO 31/XI*,
    Claudio Beccari, TUGboat Volume 18 (1997), No. 1.

[2] *Typefaces for Symbols in Scientific Manuscripts.*
    https://www.physics.nist.gov/cuu/pdf/typefaces.pdf.

[3] *On the Use of Italic and up Fonts for Symbols in Scientific Text*, I.M. Mills and W.V. Metanomski, ICTNS (Interdivisional Committee on Nomenclature and Symbols), dec 1999.

[4] *ISO 80000-2*. https://en.wikipedia.org/wiki/ISO_80000-2

[5] *The amsmath package.* Frank Mittelbach, Rainer Schöpf, Michael Downes, Davis M. Jones, David Carlisle, CTAN, v2.17b 2018/12/01.

[6] *The mathtool package.* Morten Høgholm, Lars Madsen, CTAN, v1.21 2018/01/08.

[7] *Typesetting vectors with beautiful arrow with LaTeX 2ε*. esvect package by Eddie Saudrais, CTAN, v1.3 2013/07/11.

[8] *The upgreek package for LaTeX 2ε*, Walter Schmidt, CTAN, v2.0 2003/02/12.

[9] *The fixmath package for LaTeX 2ε*, Walter Schmidt, CTAN, v0.9 2000/04/11.

[10] *isomath. Mathematical style for science and technology.* Günter Milde, CTAN, v0.6.1 04/06/2012.

[11] *The interval package.* Lars Madsen, CTAN, v0.3 2014/08/04.

[12] *L'extension pour TeX et LaTeX systeme.* Christian Tellechea, CTAN v0.32 2019/01/13.

[13] *The spalign package.* Joseph Rabinoff, CTAN, 2016/10/05.

[14] *L'extension frenchmath.* Antoine Missier, CTAN, v1.4 2019/05/22.

[15] *The Not So Short Introduction to LaTeX 2ε*. lshort package by Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schlegl, CTAN, v6.2 2018/02/28.

[16] *The LaTeX Companion.* Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, 2nd edition, Pearson Education, 2004.