Documentation

LaTeX

Mathmode - v. 1.29

Herbert Voß*

28th December 2003

Abstract

More than once people say that TeX was designed for mathematical or technical purpose. This maybe true when we remember the reasons why Donald Knuth created TeX. But nowadys there are a lot of examples where TeX was used for publications without any mathematical or technical background. Nevertheless, we have to consider, that writing publications with a lot of mathematical material is one of the important advantages of TeX and it seems that is impossible to know all existing macros and options of (LA)TeX and the several additional packages, especially `amsmath.sty`. This is the reason why I tried to collect all important facts in this paper.

Please report typos or any other comments to this documentation to voss@perce.de.

This document was written with `Kile: 1.6a (Qt: 3.1.1; KDE: 3.1.1;` http://sourceforge.net/projects/kile/) and the PDF output was build with the for Linux free available `VTeX/Linux, Version 8.34` (http://www.micropress-inc.com/linux/)

*Thanks for the feedback to: Christian Faulhammer <v-li@gmx.de>; José Luis Gómez Dans <josegomez@gmx.net>; Morten Høgholm<moho01ab@student.cbs.dk>; Angus Leeming <leeming@lyx.org>; Tim Love <tpl@eng.cam.ac.uk>; Uwe Stöhr <uwestoehr@web.de>

Contents

Part I
Standard LaTeX Mathmode

1 Introduction

The following sections describe all the math commands which are available without any additional package. Most of them also work with special packages and some of them are redefined. At first some important facts for typesetting math expressions.

2 The Inlinemode

\smallmatrix

As the name says this are always math expressions which are in a standard textline, like this one: $f(x)=\int_a^b \frac{\sin x}{x}dx$. There are no limitations for the height of the math expressions, so that the layout may be very lousy if you insert a big matrix in an inlinemode like this: $\underline{A}=\left[\begin{array}{ccc} a & b & c\\ d & e & f\\ g & h & i\end{array}\right]$. In this case it's better to use the `\smallmatrix` environment $\underline{A}=\left[\begin{smallmatrix} a & b & c\\ d & e & f\\ g & h & i\end{smallmatrix}\right]$ (see section 5 on page 20) or the displaymathmode Chapter 3 on page 10.

This style is possible with three different commands:

```
\( \sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1) \)
 $ \sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1) $
\begin{math}
   \sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)
\end{math}
```

$\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$ $\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$ $\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$

1. `\( ... \)` \(...\)
2. `\small $ ... $` $...$
3. `\begin{math}` ... `\end{math}` \begin{math} ... \end{math}

2.1 Limits

In the inline mode the limits are by default only in super or subscript mode and the fractions are always in scriptsize[1]. For example: $\int_1^\infty \frac{1}{x}dx=1$, which is not too big for the textline. You can change this with the command `\limits`, which must follows a mathoperator[2] like an integral (`\int`), a sum

\limits
\int
\lim
\prod
\sum

[1] see section 12 on page 33.
[2] To define a new operator see section 65

(\sum), a product (\prod) or a limes (\lim) But this $\int\limits_1^\infty \frac{1}{x}dx = 1$ looks not very nice in a text line when it appears between two lines, especially when there are multiline limits.[3]

2.2 \fraction command

For inlined formulas the fractions are by default in the scriptstyle (see tabular 8 on page 33), which is good for the typesetting $y = \frac{a}{b+1}$, because the linespacing is nearly the same, but not optimal, when the formula shows some important facts. There are two solutions to get a better reading: \fraction

1. choose the display mode instead of the inline mode, which is the better one;
2. set the fontstyle to displaystyle, which makes the fraction $y = \dfrac{a}{b+1}$ more readable but the linespacing increases which is always a bad solution and should only be used when the first solution makes no sense.[4]

```
$y={\displaystyle\frac{a}{b+1}}$
```

2.3 Math in \part, \chapter, \section, ... titles like $f(x) = \prod_{i=1}^{n}\left(i - \frac{1}{2i}\right)$

All commands which appear in positions like contents, index, header, ... must be robust. If you do not have any contents, index, a.s.o. you can write the mathstuff in \chapter, \section, a.s.o without any restriction. otherwise put a more \protect before these commands. The whole math expression appears in the default font shape and not in bold like the other text. Section 22.1 on page 42 describes who the math expressions can be printed also in bold.

2.4 Equation numbering

It's obvious that the numbering of inline mathstuff makes no sense!

2.5 Framed Math

With the \fbox macro everything of inline math can be framed, like the following one: $\boxed{f(x) = \prod_{i=1}^{n}\left(i - \frac{1}{2i}\right)}$.

```
\fbox{$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)$}
```

[3]For more information about limits see section 6 on page 22 or section 40 on page 63

[4]For an abbreviation see section 34 on page 57, there is a special \dfrac macro.

Parameters are the width of `\fboxsep` and `\fboxrule`, the predefined values from `latex.ltx` are:

```
\fboxsep = 3pt
\fboxrule = .4pt
```

The same is possible with the `\colorbox` $f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)$ from the color package.

```
\colorbox{yellow}{$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)$}
```

2.6 Linebreak

LaTeX can break an inline formula only when a relation symbol $(=,<,>,\ldots)$ or a binary operation symbol $(+,-,\ldots)$ exists and at least one of these symbols appears at the outer level of a formula. Thus `$a+b+c$` can be broken across lines, but `${a+b+c}$` not.

- The default: $f(x)=a_nx^n+a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\ldots+a_ix^i+a_2x^2+a_1x^1+a_0$
- The same inside a group {...}: ${f(x)=a_nx^n+a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\ldots+a_ix^i+a_2x^2+a_1x^1 +}$
- Without any symbol: $f(x)=a_n\left(a_{n-1}\left(a_{n-2}\left(\ldots\right)\ldots\right)\ldots\right)$

If it is not possible to have any mathsymbol, then split the inline formula in two or more pieces (`$...$ $...$`).

2.7 Whitespace

LaTeX defines the length `\mathsurround` with the default value of `0pt`. This length is added before and after an inlined math expression (see table 1).

default	foo $\boxed{f(x)=\int_1^\infty\frac{1}{x}dx=1}$ bar
`\rule{20pt}{\ht\strutbox}` ... `\rule{20pt}{\ht\strutbox}`	foo ▬ $\boxed{f(x)=\int_1^\infty\frac{1}{x}dx=1}$ ▬ bar
`\setlength{\mathsurround}{20pt}`	foo $\boxed{\quad f(x)=\int_1^\infty\frac{1}{x}dx=1\quad}$ bar

Table 1: Meaning of `\mathsurround`

2.8 $\mathcal{AMS}$math-stuff

None of the amsmath-functions are available in inlinemode.

3 Displaymathmode

This means, that every formula gets its own paragraph (line). There are some differences in the layout to the one from the title of 2.3.

3.1 equation environment

For example:

```
\begin{equation}
 f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)
\end{equation}}
```

The only difference to the inline-version are the delimiters `\begin{equation}` ... `\end{equation}`. There are some equivalent commands for the displaymathmode:

1. `\[ ... \]`. (see above) the short form of a displayed formula, no number

   `\[` ... `\]`

   `\begin{displaymath}`
   ...
   `\end{displaymath}`, same as

   `\begin{displaymath}`
   ...
   `\end{displaymath}`

   $$f(x) = \prod_{i=1}^{n} \left(i - \frac{1}{2i} \right)$$

   displayed, no number. Same as 1.

2. `\begin{equation} ... \end{equation}`

   `\begin{equation}` ... `\end{equation}`

   $$f(x) = \prod_{i=1}^{n} \left(i - \frac{1}{2i} \right) \tag{1}$$

   displayed, a sequential equationnumber, which may be reset when starting a new chapter/section.

   (a) There is only **one** equation number for the whole environment.

   `\nonumber`

   (b) There exists no star-version of the equation environment because `\[..\]` is the equivalent. With the tag `\nonumber` it is possible to suppress the equationnumber:

   ```
   \begin{equation}
     f(x)= [...] \right)\nonumber
   \end{equation}
   ```

3.2 `eqnarray` environment

This is by default an array with three columns and as many rows as you like. It is nearly the same as an array with a `rcl` column definition.

```
\begin{eqnarray}
...
\end{eqnarray}
```

> It's not possible to change the internal behaviour of the **`eqnarray`** environment. It's always an implicit array with **three** columns and the horizontal alignment **`right-center-left (rcl)`** and small **symbol** sizes for the middle column. All this can not be changed by the user without rewriting the whole environment in `latex.ltx`.

$$\begin{array}{rcl} \text{left} & \text{middle} & \text{right} \\ \frac{1}{\sqrt{n}} = & \frac{\sqrt{n}}{n} = & \frac{n}{n\sqrt{n}} \end{array}$$

The **`eqnarray`** environment should not be used as an array. As seen in the above example the typesetting is wrong for the middle column. The numbering of eqnarray environments is always for every row, means, that four lines get four different equation numbers (for the labels see section 3.4):

$$\begin{array}{rcl} y & = & d \qquad (2) \\ y & = & cx+d \qquad (3) \\ y & = & bx^2+cx+d \qquad (4) \\ y & = & ax^3+bx^2+cx+d \qquad (5) \end{array}$$

```latex
\begin{eqnarray}
 y & = & d\label{eq:2}\\
 y & = & cx+d\\
 y & = & bx^{2}+cx+d\\
 y & = & ax^{3}+bx^{2}+cx+d\label{eq:5}
\end{eqnarray}
```

- Toggling numbering off/on for **all** rows is possible with the starred version of eqnarray.

$$\begin{array}{rcl} y & = & d \\ y & = & cx+d \\ y & = & bx^2+cx+d \\ y & = & ax^3+bx^2+cx+d \end{array}$$

```latex
\begin{eqnarray*}
 y & = & d\label{eq:3}\\
 y & = & cx+d\\
```

```
 y & = & bx^{2}+cx+d\\
 y & = & ax^{3}+bx^{2}+cx+d\label{eq:4}
\end{eqnarray*}
```

- Toggling off/on for **single** rows is possible with the above mentioned `\nonumber` tag at the end of a row (before the newline command). For example:

$$\begin{aligned} y &= d \\ y &= cx+d \\ y &= bx^2+cx+d \\ y &= ax^3+bx^2+cx+d \qquad (6) \end{aligned}$$

```
\begin{eqnarray}
 y & = & d\nonumber \\
 y & = & cx+d\nonumber \\
 y & = & bx^{2}+cx+d\nonumber \\
 y & = & ax^{3}+bx^{2}+cx+d
\end{eqnarray}
```

3.2.1 Short commands

It is possible to define short commands for the `eqnarray` environment

```
\makeatletter
\newcommand{\be}{%
    \begingroup
%    \setlength{\arraycolsep}{2pt}
    \eqnarray%
 \@ifstar{\nonumber}{}%
}
\newcommand{\ee}{\endeqnarray\endgroup}
\makeatother
```

Now you can wrote the whole equation as

```
\be
f(x) & = & \int\frac{\sin x}{x}dx
\ee
```

$$f(x) = \int \frac{\sin x}{x} dx \qquad (7)$$

or, if you do not want to have a numbered equation as

```
\be*
f(x) & = & \int\frac{\sin x}{x}dx
\ee
```

$$f(x) = \int \frac{\sin x}{x} dx$$

3.3 Equation numbering

`\nonumber`

For all equations which can have one or more equation numbers (for every line/row) the numbering for the whole equation can be disabled with switching from the star to the unstarred version. This is still for the whole formula and doesn't works for single rows. In this case use the `\nonumber` tag.

- This doc is written with the article-class, which counts the equations continously over all parts/sections. You can change this behaviour in different ways (see the following subsections).
- In standard LaTeX it is a problem with too long equations and the equation number, which may be printed over the equation. In this case use the **amsmath** package, where the number is set above or below of a too long equation (see equation 28 on page 26).
- For counting subequations see section 38.1 on page 61.

3.3.1 Changing the Style

\theequation

With the beginning of Section 25.2 on page 47 the counting changes from „40“ into the new style „II-47“. The command sequence is

```
\renewcommand\{\theequation}{%
 \thepart-\arabic{equation}%
}
```

See section 38 on page 60 for the amsmath command.

3.3.2 Resetting a Counter Style

Removing a given reset is possible with the package **remreset**.[5] write into the preamble

`\@removefromreset`

```
\makeatletter
\@removefromreset{equation}{section}
\makeatother
```

or anywhere in the text.

Now the equation counter is no longer reset when a new section starts. You can see this after section 28 on page 52.

[5] CTAN://macros/latex/contrib/supported/carlisle/remreset.sty

3.3.3 Equation numbers on the left side

Choose package `leqno`[6] or have a look at your document class, if such an option exists.

3.3.4 Changing the equation number style

The number style can be changed with a redefinition of

```
\def\@eqnnum{{\normalfont \normalcolor (\theequation)}}
```

For example: if you want the numbers not in parentheses write

```
\makeatletter
\def\@eqnnum{{\normalfont \normalcolor \theequation}}
\makeatother
```

For `amsmath` there is another macro, see section38 on page 60.

3.3.5 More than one equation counter

You can have more than the default equation counter. With the following code you can easily toggle between roman and arabic equation counting.

```
%code by Heiko Oberdiek
\makeatletter
%Roman counter
\newcounter{roem}
\renewcommand{\theroem}{\roman{roem}}

% save the original counter
\newcommand{\c@org@eq}{}
\let\c@org@eq\c@equation
\newcommand{\org@theeq}{}
\let\org@theeq\theequation

%\setroem sets roman counting
\newcommand{\setroem}{
  \let\c@equation\c@roem
  \let\theequation\theroem}

%\setarab the arabic counting
\newcommand{\setarab}{
  \let\c@equation\c@org@eq
  \let\theequation\org@theeq}
\makeatother
```

[6] CTAN://macros/latex/unpacked/

The following examples show how it works:

$$f(x) = \int \sin x dx \tag{8}$$
$$g(x) = \int \frac{1}{x} dx \tag{9}$$

$$F(x) = -\cos x \tag{i}$$
$$G(x) = \ln x \tag{ii}$$

$$f'(x) = sinx \tag{10}$$
$$g'(x) = \frac{1}{x} \tag{11}$$

There can be references to these equations in the usual way, like eq.8, 11 and the roman one eq.ii .

```
\begin{align}
f(x)&=\int\sin x dx\label{eq:arab1}\\
g(x)&=\int\frac{1}{x}dx
\end{align}
%
\setroem
%
\begin{align}
F(x)&=-\cos x\\
G(x)&=\ln x\label{eq:rom1}
\end{align}
%
\setarab
%
\begin{align}
f\prime (x)&=sin x\\
g\prime (x)&=\frac{1}{x}\label{eq:arab2}
\end{align}
```

3.4 Labels

Every numbered equation can have a label to which a reference is possible.

- There is one restriction for the label names, they cannot include one of LaTeX's command characters.[7]
- The label names are replaced by the equation number.

\tag

If you do not want a reference to the equation number but to an own defined name then use the $\mathcal{A}\mathcal{M}\mathcal{S}$math command `\tag...`, which is described in section(39 on page 62).

[7] $ _ ^ \ & % { }

3.5 Frames

Similiar to the inline mode, displayed equations can also be framed with the `\fbox` command, like equation 12. The only difference is the fact, that the equation must be packed into a parbox or minipage. It is nearly the same for a colored box, where the `\fbox{...}` has to be replaced with `\colorbox{yellow}{...}`. The package `color.sty` must be loaded and important the `calc.sty` package to get a correct boxwidth.

$$\boxed{f(x)=\int_1^{\inf}\dfrac{1}{x}dx=1 \qquad (12)}$$

```
\noindent\fbox{\parbox{\linewidth-2\fboxsep-2\fboxrule}{%
\begin{equation}\label{eq:frame0}
 f(x)=\int_1^{\inf}\dfrac{1}{x}dx=1
\end{equation}%
}}
```

If the equation number should not be part of the frame, then it is a bit complicated. There is one tricky solution, which puts an unnumbered equation just beside an empty numbered equation. The `\hfill` is only useful for placing the equation number right aligned, which is not the default. The following three equations 13-3.5 are the same, only the second one written with the `\myMathBox` macro which has the border and background color as optional arguments with the defaults `white` for background and `black` for the frame. If there is only one optional argument, then it is still the one for the frame color (3.5).

```
\makeatletter
\def\myMathBox{\@ifnextchar[{\my@MBoxi}{\my@MBoxi[black]}}
\def\my@MBoxi[#1]{\@ifnextchar[{\my@MBoxii[#1]}{\my@MBoxii[#1][white
   ]}}
\def\my@MBoxii[#1][#2]#3{%
  \par\noindent%
  \fcolorbox{#1}{#2}{%
     \parbox{\linewidth-\labelwidth-2\fboxrule-2\fboxsep}{#3}%
  }%
  \parbox{\labelwidth}{%
    \begin{eqnarray}\end{eqnarray}%
  }%
  \par%
}
\makeatother
```

$$f(x)=x^2+x \tag{13}$$

$$\boxed{f(x)=x^2+x} \qquad (14)$$

$$f(x) = x^2 + x \tag{15}$$

$$f(x) = x^2 + x \tag{16}$$

```
\begin{equation}\label{eq:frame2}f(x)=x^2 +x\end{equation}
\myMathBox[red]{\[\label{eq:frame3}f(x)=x^2 +x\]}
\myMathBox[red][yellow]{\[\label{eq:frame4}f(x)=x^2 +x\]}
\myMathBox{\[\label{eq:frame5}f(x)=x^2 +x\]}
```

If you are using the **amsmath** package, then try the solutions from section 44 on page 67.

4 array environment

\begin{array}
...
\end{array}

This is simply the same as the eqnarray environment only with the possibility of variable rows **and** columns and the fact, that the whole formula has only **one** equation number and that the `array` environment can only be part of another math environment, like `equation` or `displaymath`.

$$\left.\begin{array}{ccccrr} \text{a)} & & y & = & c & (constant)\\ \text{b)} & & y & = & cx+d & (linear)\\ \text{c)} & & y & = & bx^{2}+cx+d & (square)\\ \text{d)} & & y & = & ax^{3}+bx^{2}+cx+d & (cubic) \end{array}\right\} \text{Polynomes} \tag{17}$$

```
\begin{equation}
\left.%
\begin{array}{ccccrr}
 \textrm{a}) & & y & = & c & (constant)\\
 \textrm{b}) & & y & = & cx+d & (linear)\\
 \textrm{c}) & & y & = & bx^{2}+cx+d & (square)\\
 \textrm{d}) & & y & = & ax^{3}+bx^{2}+cx+d & (cubic)
\end{array}%
\right\} \textrm{Polynomes}
\end{equation}
```

The horizontal alignment of the columns is the same than the one from the `tabular` environment.

4.1 Arrays with Delimiters (delarray.sty)

Package `delarray.sty`[8] supports different delimiters which are defined together with the beginning of an array:

[8] CTAN://macros/latex/required/tools/delarray.dtx

```
\begin{array}<delLeft>{cc}<delRight>
...
```

defines an array with two centered colums and the delimters „`<delLeft><delRight>`“, f.ex. „`()`“.

```
\[
A=\begin{array}({cc})
 a & b\\
 c & d
\end{array}
\]
```

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

`delarray.sty` expects a pair of delimiters. If you need only one (like the cases-structure) then use the dot for an „empty“ delimiter, f.ex.

```
\[
A=\begin{array}{{cc}.
 a & b\\
 c & d
\end{array}
\]
```

$$A = \left\{ \begin{array}{cc} a & b \\ c & d \end{array} \right.$$

which is a useful command for a cases structure without the `amsmath` package, which is described in the amsmath part.

4.2 Cases Structure

If you do not want to use the amsmath package then write your own cases-structure with the `array` environment. For the above example do it in the following way:

```
\[x = \left\{%
   \begin{array}{l>{\raggedright}p{.5\textwidth}}%
      0 & if A=...\tabularnewline
      1 & if B=...\tabularnewline
      x & this runs with as much text as you like, %
          because an automatic linebreak is given with %
          an raggedright text. Without this %
          \raggedright command, you'll get a formatted %
          text, like the following one ...
    \end{array}%
  \right. %
\]
```

$$x = \left\{ \begin{array}{ll} 0 & \text{if A=...} \\ 1 & \text{if B=...} \\ x & \text{this runs with as much text as you like, but without an ...} \end{array} \right. \tag{18}$$

It is obvious, that we need a `\parbox` if the text is longer than the possible linewidth.

$$x = \left\{ \begin{array}{ll} 0 & \text{if A=...} \\ 1 & \text{if B=...} \\ x & \parbox{0.5\columnwidth}{this runs with as much text as you like, because an automatic linebreak is given with an raggedright text. Without this command, you'll get a formatted text, like the following one ... but with a parbox ... it works} \end{array} \right.$$

```
\[
x = \left\{%
   \begin{array}{l>{\raggedright}p{.5\textwidth}}%
      0 & if A=...\tabularnewline
      1 & if B=...\tabularnewline
      x & \parbox{0.5\columnwidth}{this runs with as much text as
          you like, %
          because an automatic linebreak is given with %
          an raggedright text. Without this %
          \raggedright command, you'll get a formatted %
          text, like the following one ... but with a parbox ... it
              works}
    \end{array}%
  \right. %
\]
```

4.3 arraycolsep

\arraycolsep

All the foregoing math environments use the array to typset the math expression. The predefined separation between two columns is the length `\arraycolsep`, which is set by nearly all document classes to `5pt`, which seems to be too big. The following equation is typeset with the default value and the second one with `\arraycolsep=1.4pt`

$$\boxed{f(x)} \quad \boxed{=} \quad \boxed{\int \frac{\sin x}{x} dx}$$

$$\boxed{f(x)}\boxed{=}\boxed{\int \frac{\sin x}{x} dx}$$

If this modification should be valid for all arrays/equations, then write it into the preamble, otherwise put it into a group or define your own environment as done in section 3.2.1 on page 12.

```
\bgroup
\arraycolsep=1.4pt
\begin{eqnarray}
f(x) & = & \int\frac{\sin x}{x}dx
\end{eqnarray}
\egroup
```

```
\makeatletter
\newcommand{\be}{%
    \begingroup
    \setlength{\arraycolsep}{1.4pt}
[ ... ]
```

5 Matrix

\matrix
\bordermatrix
\smallmatrix

TeX knows two macros and LaTeX one more for typesetting a matrix:

$$\begin{matrix} A & B & C \\ d & e & f \\ 1 & 2 & 3 \end{matrix}$$

```
\matrix{%
 A & B & C \cr
 d & e & f \cr
 1 & 2 & 3 \cr%
}
```

$$\begin{array}{c} \\ 0 \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{matrix} 0 & 1 & 2 \end{matrix} \\ \begin{pmatrix} A & B & C \\ d & e & f \\ 1 & 2 & 3 \end{pmatrix} \end{array}$$

```
\bordermatrix{
   & 0 & 1 & 2 \cr
 0 & A & B & C \cr
 1 & d & e & f \cr
 2 & 1 & 2 & 3 \cr
}
```

$\begin{smallmatrix} A & B & C \\ d & e & f \\ 1 & 2 & 3 \end{smallmatrix}$

```
\begin{smallmatrix}%
 A & B & C \\
 d & e & f \\
 1 & 2 & 3
\end{smallmatrix}
```

The first two macros are listed here for some historical reason, because the `array` or especially the `amsmath` package offer the same or better macros/environments. Nevertheless it is possible to redefine the `bordermatrix` macro to get other parentheses and a star version which takes the left top part as matrix:

$$\begin{array}{cc} & \begin{array}{cc}1 & 2\end{array}\\ \begin{array}{c}1\\2\\3\end{array} & \begin{pmatrix}x1 & x2\\ x3 & x4\\ x5 & x6\end{pmatrix}\end{array} \qquad \begin{array}{cc} & \begin{array}{cc}1 & 2\end{array}\\ \begin{array}{c}1\\2\\3\end{array} & \begin{bmatrix}x1 & x2\\ x3 & x4\\ x5 & x6\end{bmatrix}\end{array} \qquad \begin{array}{cc} & \begin{array}{cc}1 & 2\end{array}\\ \begin{array}{c}1\\2\\3\end{array} & \left\{\begin{matrix}x1 & x2\\ x3 & x4\\ x5 & x6\end{matrix}\right\}\end{array} \tag{19}$$

$$\begin{array}{cc} \begin{pmatrix}x1 & x2\\ x3 & x4\\ x5 & x6\end{pmatrix} & \begin{array}{c}1\\2\\3\end{array}\\ \begin{array}{cc}1 & 2\end{array} & \end{array} \qquad \begin{array}{cc} \begin{bmatrix}x1 & x2\\ x3 & x4\\ x5 & x6\end{bmatrix} & \begin{array}{c}1\\2\\3\end{array}\\ \begin{array}{cc}1 & 2\end{array} & \end{array} \qquad \begin{array}{cc} \left\{\begin{matrix}x1 & x2\\ x3 & x4\\ x5 & x6\end{matrix}\right\} & \begin{array}{c}1\\2\\3\end{array}\\ \begin{array}{cc}1 & 2\end{array} & \end{array} \tag{20}$$

```
\bordermatrix{%
  & 1 & 2 \cr
1 & x1 & x2 \cr
2 & x3 & x4 \cr
3 & x5 & x6
}
```

```
\bordermatrix[{[]}]{%
  & 1 & 2 \cr
1 & x1 & x2 \cr
2 & x3 & x4 \cr
3 & x5 & x6
}
```

```
\bordermatrix[\{\}]{%
  & 1 & 2 \cr
1 & x1 & x2 \cr
2 & x3 & x4 \cr
3 & x5 & x6
}
```

```
\bordermatrix*{%
 x1 & x2 & 1 \cr
 x3 & x4 & 2 \cr
 x5 & x6 & 3 \cr
  1 & 2
}
```

```
\bordermatrix*[{[]}]{%
 x1 & x2 & 1 \cr
 x3 & x4 & 2 \cr
 x5 & x6 & 3 \cr
  1 & 2
}
```

```
\bordermatrix*[\{\}]{%
 x1 & x2 & 1 \cr
 x3 & x4 & 2 \cr
 x5 & x6 & 3 \cr
  1 & 2
}
```

There is now an optional argument for the parenthesis with `()` as the default one. To get such a behaviour, write into the preamble:

```
\makeatletter
\newif\if@borderstar
\def\bordermatrix{\@ifnextchar*{%
 \@borderstartrue\@bordermatrix@i}{\@borderstarfalse\@bordermatrix@i*}%
}
\def\@bordermatrix@i*{\@ifnextchar[{\@bordermatrix@ii}{\@bordermatrix@ii[()
    ]}}
\def\@bordermatrix@ii[#1]#2{%
\begingroup
 \m@th\@tempdima8.75\p@\setbox\z@\vbox{%
  \def\cr{\crcr\noalign{\kern 2\p@\global\let\cr\endline }}%
  \ialign {$###$\hfil\kern 2\p@\kern\@tempdima & \thinspace %
   \hfil $###$\hfil && \quad\hfil $###$\hfil\crcr\omit\strut %
   \hfil\crcr\noalign{\kern -\baselineskip}#2\crcr\omit %
   \strut\cr}}%
 \setbox\tw@\vbox{\unvcopy\z@\global\setbox\@ne\lastbox}%
 \setbox\tw@\hbox{\unhbox\@ne\unskip\global\setbox\@ne\lastbox}%
 \setbox\tw@\hbox{%
  $\kern\wd\@ne\kern -\@tempdima\left\@firstoftwo#1%
```

```
   \if@borderstar\kern2pt\else\kern -\wd\@ne\fi%
  \global\setbox\@ne\vbox{\box\@ne\if@borderstar\else\kern 2\p@\fi}%
  \vcenter{\if@borderstar\else\kern -\ht\@ne\fi%
   \unvbox\z@\kern-\if@borderstar2\fi\baselineskip}%
  \if@borderstar\kern-2\@tempdima\kern2\p@\else\,\fi\right\@secondoftwo#1 $%
 }\null \;\vbox{\kern\ht\@ne\box\tw@}%
\endgroup
}
\makeatother
```

The `matrix` macro can not be used together with the `amsmath` package, it redefines this macro (see section 31 on page 55). The `smallmatrix` environment makes some sense in the inline mode to decrease the line height.

6 Multiple Limits

`\atop`

For general information about limits read section 2.1 on page 7. With the `\atop` command multiple limits for a sum or prod are possible. The syntax is:

```
{above \atop below}
```

which is nearly the same than a fraction without a rule. This can be enhanced to `a\atop b\atop c` and so on. For equation 21 do the following steps:

```
\begin{equation}
\sum_{{1\le j\le p\atop {%
  {1\le j\le q\atop 1\le k\le r}}}%
 }a_{ij}b_{jk}c_{ki}
\end{equation}
```

$$\sum_{{1\le j\le p\atop {1\le j\le q\atop 1\le k\le r}}} a_{ij}b_{jk}c_{ki} \tag{21}$$

There are other solutions to get multiple limits, f.ex. an array, which is not the best solution because the space between the lines is too big. The amsmath-package provides several commands for limits (section 40) and the `\underset` and `\overset` commands (see section 46).

6.1 Problems

$$\boxed{\sum_{{1\le j\le p\atop {1\le j\le q\atop 1\le k\le r}}}}\boxed{a_{ij}b_{jk}c_{ki}} \tag{22}$$

The equation 22 shows that the horizontal alignment is not optimal, because the math expression on the right follows at the end of the limits which are a unit together with the sum symbol. There is an elegant solution with **amsmath.sty**, described in subsection 40.2 on page 63. If you do not want to use **amsmath.sty**, then use `\makebox`. But there is a problem when the general fontsize is increased, `\makebox` knows nothing about the actual math font size. Equation 23a shows the effect and equation 23b the view without the boxes.

$$\boxed{\sum_{\substack{1\le j\le p\\1\le j\le q\\1\le k\le r}}}\boxed{a_{ij}b_{jk}c_{ki}} \quad (23a) \qquad \sum_{\substack{1\le j\le p\\1\le j\le q\\1\le k\le r}} a_{ij}b_{jk}c_{ki} \quad (23b)$$

```
\begin{equation}
\sum_{\makebox[0pt]{$%
 {{\scriptscriptstyle 1\le j\le p\atop {%
 {1\le j\le q\atop 1\le k\le r}}}}%
  $}}a_{ij}b_{jk}c_{ki}
\end{equation}
```

7 Roots

The square root `\sqrt` is is the default for LaTeX and the n-th root can be inserted with the optional parameter `\sqrt[n]....` . \sqrt

`\sqrt{x}`	$\sqrt{x}$
`\sqrt[3]{x}`	$\sqrt[3]{x}$

There is a different typesetting in roots. Equations 24 has different heights for the roots, whereas equation 25 has the same one. This is possible with the `\vphantom` command, which reserves the vertical space (without a horizontal one) of the parameter height. \vphantom

$$\sqrt{a}\,\sqrt{T_{\mathrm{r}}}\,\sqrt{2\alpha k_{\mathrm{B}}T^{i}} \tag{24}$$

```
\begin{equation}
 \sqrt{a}\,%
 \sqrt{T_{\mathrm{r}}}\,%
 \sqrt{2\alpha k_{\mathrm{B}}T^{i}}\label{eq:root1}
\end{equation}
```

$$\sqrt{a_{\vphantom{B}}^{\vphantom{i}}}\,\sqrt{T_{\mathrm{r}\vphantom{B}}^{\vphantom{i}}}\,\sqrt{2\alpha k_{\mathrm{B}}T^{i}} \tag{25}$$

```
\begin{equation}
 \sqrt{a_{\vphantom{B}}^{\vphantom{i}}}\,%
 \sqrt{T_{\mathrm{r}\vphantom{B}}^{\vphantom{i}}}\,%
 \sqrt{2\alpha k_{\mathrm{B}}T^{i}}\label{eq:root2}
\end{equation}
```

The typesetting looks much more better, especially when the formula has different roots in a row, like equation 24 on the page before. Using `amsmath.sty` with the `\smash` command[9] gives some more possibilities for typesetting of roots (see section 35 on page 59).

8 Brackets, Braces and parentheses

This is one of the major problems inside the mathmode, because there is often a need for different brackets, braces and parentheses in different size. At first we had to admit, that there is a difference between the characters „`()[]/\` $\{\}\ |\ \|\ \lfloor\rfloor\ \lceil\rceil\ \langle\rangle\ \uparrow\Uparrow\ \downarrow\Downarrow\ \updownarrow\Updownarrow$“ and their use as an argument of the `\left` and `\right` command, where LaTeX stretches the size in a way that all between the pair of left and right parentheses is smaller than the parentheses. In some cases[10] it may be useful to choose a fixed height, which is possible with the `\big`-series. Instead of writing `\leftX` or `\rightX` one of the following commands can be chosen:

`\leftX` `\rightX`

default $()[]/\backslash\{\}|\|\lfloor\rfloor\lceil\rceil\langle\rangle\uparrow\Uparrow\downarrow\Downarrow\updownarrow\Updownarrow$

\bigX $\big(\big)\ \big[\big]\ \big/\big\backslash\big\{\big\}\ \big|\ \big\|\ \big\lfloor\big\rfloor\ \big\lceil\big\rceil\ \big\langle\big\rangle\ \big\uparrow\big\Uparrow\ \big\downarrow\big\Downarrow\ \big\updownarrow\big\Updownarrow$

\BigX $\Big(\Big)\ \Big[\Big]\ \Big/\Big\backslash\Big\{\Big\}\ \Big|\ \Big\|\ \Big\lfloor\Big\rfloor\ \Big\lceil\Big\rceil\ \Big\langle\Big\rangle\ \Big\uparrow\Big\Uparrow\ \Big\downarrow\Big\Downarrow\ \Big\updownarrow\Big\Updownarrow$

`\biggX` $\bigg(\bigg)\ \bigg[\bigg]\ \bigg/\bigg\backslash\bigg\{\bigg\}\ \bigg|\ \bigg\|\ \bigg\lfloor\bigg\rfloor\ \bigg\lceil\bigg\rceil\ \bigg\langle\bigg\rangle\ \bigg\uparrow\bigg\Uparrow\ \bigg\downarrow\bigg\Downarrow\ \bigg\updownarrow\bigg\Updownarrow$

\BiggX $\Bigg(\Bigg)\ \Bigg[\Bigg]\ \Bigg/\Bigg\backslash\Bigg\{\Bigg\}\ \Bigg|\ \Bigg\|\ \Bigg\lfloor\Bigg\rfloor\ \Bigg\lceil\Bigg\rceil\ \Bigg\langle\Bigg\rangle\ \Bigg\uparrow\Bigg\Uparrow\ \Bigg\downarrow\Bigg\Downarrow\ \Bigg\updownarrow\Bigg\Updownarrow$

`\bigX` `\BigX` `\biggX` `\BiggX`

Only a few commands can be written in a short form like `\big(`. The „X“ has to be replaced with one of the following characters or commands from table 3, which shows the parentheses character, its code for the use with one of the „big“ commands and an example with the code for that.

`\biglX` `\bigrX`

- There exists for all commands a left/right version `\bigl`, `\bigr`, `\Bigl` and so on, which makes only sense when writing things like:

$$\Big)\times\frac{a}{b}\times\Big(\tag{26}$$

$$\Bigr)\times\frac{a}{b}\times\Bigl(\tag{27}$$

[9] The `\smash` command exists also in LaTeX but without an optional argument, which makes the use for roots possible.

[10] See section 8.1.1 on page 26 for example.

```
\begin{align} \biggl)\times \frac{a}{b} \times\biggr( \end{align}
\begin{align} \bigg)\times \frac{a}{b} \times\bigg( \end{align}
```

LaTeX takes the `\biggl)` as a mathopen symbol, which has by default another horizontal spacing.

- In addition to the above additional commands there exists some more: `\bigm`, `\Bigm`, `\biggm` and `\Biggm`, which work as the standard ones (without the addtional „m“) but add some more horizontal space between the delimiter and the formula before and after (see table 2). `\bigmX` `\bigmX`

$3\bigg\|a^2-b^2-c^2\bigg\|+2$	`3\bigg|a^2-b^2-c^2\bigg|+2`
$3\biggm\|a^2-b^2-c^2\biggm\|+2$	`3\biggm|a^2-b^2-c^2\biggm|+2`

Table 2: Difference between the default `\bigg` and the `\biggm` Command

Char.	Code	Example	Code
( )	( )	$3\Big(a^2+b^{c^2}\Big)$	`3\Big( a^2+b^{c^2}\Big)`
[ ]	[ ]	$3\Big[a^2+b^{c^2}\Big]$	`3\Big[ a^2+b^{c^2}\Big]`
/ \	`\backslash`	$3\Big/ a^2+b^{c^2}\Big\backslash$	`3\Big/ a^2+b^{c^2}\Big\backslash`
{ }	\{ \}	$3\Big\{ a^2+b^{c^2}\Big\}$	`3\Big\{ a^2+b^{c^2}\Big\}`
$\vert$ $\Vert$	$\vert$ \Vert	$3\Big\vert a^2+b^{c^2}\Big\Vert$	`3\Big| a^2+b^{c^2}\Big\Vert`
$\lfloor$ $\rfloor$	`\lfloor` `\rfloor`	$3\Big\lfloor a^2+b^{c^2}\Big\rfloor$	`3\Big\lfloor a^2+b^{c^2}\Big\rfloor`
$\lceil$ $\rceil$	\lceils \rceil	$3\Big\lceil a^2+b^{c^2}\Big\rceil$	`3\Big\lceil a^2+b^{c^2}\Big\rceil`
$\langle$ $\rangle$	\langle \rangle	$3\Big\langle a^2+b^{c^2}\Big\rangle$	`3\Big\langle a^2+b^{c^2}\Big\rangle`
$\uparrow$ $\Uparrow$	\uparrow \Uparrow	$3\Big\uparrow a^2+b^{c^2}\Big\Uparrow$	`3\Big\uparrow a^2+b^{c^2}\Big\Uparrow`
$\downarrow$ $\Downarrow$	\downarrow \Downarrow	$3\Big\downarrow a^2+b^{c^2}\Big\Downarrow$	`3\Big\downarrow a^2+b^{c^2}\Big\Downarrow`

Char.	Code	Example	Code
$\updownarrow$ $\Updownarrow$	\updownarrow \Updownarrow	$3\Big\updownarrow a^2+b^{c^2}\Big\Updownarrow$	`3\Big\updownarrow` `a^2+b^{c^2}\Big\Updownarrow`

Table 3: Use of the different parentheses for the "big" commands

8.1 Examples

8.1.1 Braces over several lines

The following equation in the single line mode looks like

$$\frac{1}{2}\Delta(f_{ij}f^{ij})=2\left(\sum_{i<j}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}[2\nabla_{i}R_{jk}-\nabla_{k}R_{ij}]\right) \tag{28}$$

and is too long for the text width and the equation number has to be placed under the equation. [11] With the array environment the formula can be split in two smaller pieces:

$$\begin{array}{r}
\frac{1}{2}\Delta(f_{ij}f^{ij})=2\left(\displaystyle\sum_{i<j}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\right.\\
\left.+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}[2\nabla_{i}R_{jk}-\nabla_{k}R_{ij}]\right)
\end{array} \tag{29}$$

It's obvious that there is a problem with the right closing parentheses. because of the two pairs „`\left( ... \right.`“ and „`\left. ... \right)`“ they have a different size because every pair does it in its own way. Using the `\Bigg` command changes this into a better typesetting:

$$\begin{array}{r}
\frac{1}{2}\Delta(f_{ij}f^{ij})=2\Bigg(\displaystyle\sum_{i<j}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\\
+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}[2\nabla_{i}R_{jk}-\nabla_{k}R_{ij}]\Bigg)
\end{array} \tag{30}$$

```
\begin{equation}
\begin{array}{r}
 \frac{1}{2}\Delta(f_{ij}f^{ij})=2\Bigg({\displaystyle
 \sum_{i<j}}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}%
 \nabla_{j}\nabla_{i}(\Delta f)+\\
```

[11] In standard LaTeX the equation number is printed over too long formulas. Only amsmath.sty puts it over (left numbers) or under (right numbers) the formula.

```
 +\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}[2
 \nabla_{i}R_{jk}-\nabla_{k}R_{ij}]\Bigg)
\end{array}
\end{equation}
```

Section 29 on page 54 shows another solution for getting the right size for parentheses when breaking the equation in smaller pieces.

8.1.2 Middle Bar

See section 52 on page 72 for examples and the use of package `braket.sty`.

8.2 New Delimiters

The default delimiters are defined in the file `fontmath.ltx` which is stored in general in `[TEXMF]/tex/latex/base/fontmath.ltx`. If we need for example a thicker vertical symbol than the existing `\verb` symbol we can define in the preamble:

```
\DeclareMathDelimiter{\Norm}
   {\mathord}{largesymbols}{"3E}{largesymbols}{"3E}
```

The character number $3E_{16}$ (decimal 62) from the `cmex10` font is the small thick vertical rule. Now the new delimiter `\Norm` can be used in the usual way:

$\left\| *BLA* \right\|$

$\left\| \dfrac{*BLA*}{*BLUB*} \right\|$

```
$\left\Norm *BLA* \right\Norm$

$\left\Norm \dfrac{*BLA*}{*BLUB*} \right\Norm$
```

8.3 Problems with parenthesis

`\delimitershortfall`
`\delimiterfactor`

It is obvious that the following equation has not the right size of the parenthesis.

$$\int_\gamma F'(z)dz = \int_\alpha^\beta F'\left(\gamma(t)\right)\cdot\gamma'(t)dt$$

```
\[
\int_\gamma F'(z) dz = \int_\alpha^\beta
F' \left(\gamma (t) \right) \cdot \gamma '(t) dt
\]
```

The problem is that TeX controlls the height of the parenthesis with `\delimitershortfall` and `\delimiterfactor`, with the default values

```
\delimitershortfall=5pt
\delimiterfactor=901
```

`\delimiterfactor/1000` is the relative size of the parenthesis for a given formula environment. They could be of `\delimitershortfall` too short. These values are valid at the end of the formula, the best way is to set them straight before the math environment or global for all in the preamble.

$$\int_\gamma F'(z)dz = \int_\alpha^\beta F'\left(\gamma(t)\right) \cdot \gamma'(t)dt$$

```
{\delimitershortfall=-1pt
\[
\int_\gamma F'(z) dz = \int_\alpha^\beta
F' \left(\gamma (t) \right) \cdot \gamma '(t) dt
\]}
```

9 Text in Mathmode

Standard text in mathmode should be written in upright shape and not in the italic one which is reserved for the variable names: $\boxed{I\ am\ \text{text}\ inside\ math}$. or one of table 7 on page 31. There are different ways to write text inside math.

`\textstyle`
`\mbox`
`\mathrm`

- `\mathrm`. It is like mathmode (no spaces), but in upright mode
- `\textrm`. Upright mode with printed spaces (real textmode)
- `\mbox`. The font size is still the one from `\textstyle` (see section 12 on page 33), so that you have to place additional commands when you use `\mbox` in a super- or subscript for limits.

Additional commands for text inside math are provided by amsmath (see section 42 on page 65).

10 Font commands

10.1 Old-Style Font commands

Should never be used, but are still present and supported by LaTeX. The default syntax for the old commands is

```
{\XX test}
```

Table 4 shows what for the XX have to be replaced. The major difference to the new style is that these `\XX` are toggling the actual mathmode into the „XX“ one, whereas the new commands starts a group which switches at its end back to the mode before.

`\bf`	**test**	`\cal`	$\mathcal{TEST}$	`\it`	*test*	`\rm`	test	`\tt`	`test`

Table 4: Old Font style commands

10.2 New-Style Font commands

`\mathrm`
`\mathfrak`
`\mathcal`
`\mathsf`
`\mathbb`
`\mathtt`
`\mathit`
`\mathbf`

The default syntax is

```
\mathXX{test}
```

Table 5 shows what for the XX have to be replaced. See section 55 on page 75 for additional packages.

Command	Test
default	$ABCDEFGHIJKLMNOPQRSTUVWXYZ$
	$abcdefghijklmnopqrstuvwxyz$
\mathfrak	$\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
	$\mathfrak{abcdefghijklmnopqrstuvwxyz}$
\mathcal[a]	$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
\mathsf	$\mathsf{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
	$\mathsf{abcdefghijklmnopqrstuvwxyz}$
\mathbb[a]	$\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
\mathtt	$\mathtt{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
	$\mathtt{abcdefghijklmnopqrstuvwxyz}$
\mathit	$\mathit{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
	$\mathit{abcdefghijklmnopqrstuvwxyz}$
\mathrm	$\mathrm{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
	$\mathrm{abcdefghijklmnopqrstuvwxyz}$
\mathbf	$\mathbf{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
	$\mathbf{abcdefghijklmnopqrstuvwxyz}$

Table 5: Fonts in Mathmode

11 Spaces

11.1 Math Typesetting

`\thinmuskip`
`\medmuskip`
`\thickmuskip`

LaTeX defines the three math lengthes[12] with the following values[13]:

[12] For more information see: http://www.tug.org/utilities/plain/cseq.html
[13] see `fontmath.ltx`

```
\setlength{\thinmuskip}{3mu}
\setlength{\medmuskip}{4mu plus 2mu minus 4mu}
\setlength{\thickmuskip}{5mu plus 5mu}
```

where `mu` is the abbreviation for `math unit`.

$$1\mathrm{mu} = \frac{1}{18}\mathrm{em}$$

default	$\boxed{f(x)}\boxed{=}\boxed{x^2 + 3x_0 \cdot \sin x}$
`\thinmuskip=0mu`	$\boxed{f(x)}\boxed{=}\boxed{x^2 + 3x_0 \cdot \sin\!x}$
`\medmuskip=0mu`	$\boxed{f(x)}\boxed{=}\boxed{x^2{+}3x_0{\cdot}\sin x}$
`\thickmuskip=0mu`	$\boxed{f(x)}\boxed{=}\boxed{x^2 + 3x_0 \cdot \sin x}$
all set to zero	$\boxed{f(x)}\boxed{=}\boxed{x^2{+}3x_0{\cdot}\sin\!x}$

Table 6: The meaning of the math spaces

These lenghts can have all glue and are used for the horizontal spacing in math expressions where TEX puts spaces between symbols and operators. The meaning of these different horizontal skips is shown in the table 6. For a better typesetting LATEX inserts different spaces between the symbols.

`\thinmuskip` space between ordinary and operator atoms

`\medmuskip` space between ordinary and binary atoms in display and text styles

`\thickmuskip` space between ordinary and relation atoms in display and text styles

11.2 Additional Horizontal Spacing

`\thinspace`
`\medspace`
`\thickspace`
`\negthinspace`
`\negmedspace`
`\negthickspace`

LaTeX defines the following short commands:

```
\def\>{\mskip\medmuskip}
\def\;{\mskip\thickmuskip}
\def\!{\mskip-\thinmuskip}
```

In mathmode there is often a need for additional tiny spaces between variables, f.ex. $L\frac{di}{dt}$ written with a tiny space between L and $\frac{di}{dt}$ looks nicer: $L\,\frac{di}{dt}$. Table 7 shows a list of all commands for horizontal space which can be used in mathmode. The „space“ is seen „between“ the boxed a and b. For all examples a is `\boxed{a}` and b is `\boxed{b}`. The short forms for some spaces may cause problems with other packages. In this case use the long form of the commands.

`\hspace`
`\hphantom`
`\kern`

Positive Space		Negative Space	
`$ab$`	$\boxed{a}\boxed{b}$		
`$a b$`	$\boxed{a}\boxed{b}$		
`$a\ b$`	$\boxed{a}\ \boxed{b}$		
`$a\mbox{\textvisiblespace}b$`	$\boxed{a}$␣$\boxed{b}$		
`$a\,b$` (`$a\thinspace b$`)	$\boxed{a}\,\boxed{b}$	`$a\! b$`	$\boxed{a}\!\boxed{b}$
`$a\: b$` (`$a\medspace b$`)	$\boxed{a}\:\boxed{b}$	`$a\negmedspace b$`	$\boxed{a}\negmedspace\boxed{b}$
`$a\; b$` (`$a\thickspace b$`	$\boxed{a}\;\boxed{b}$	`$a\negthickspace b$`	$\boxed{a}\negthickspace\boxed{b}$
`$a\quad b$`	$\boxed{a}\quad\boxed{b}$		
`$a\qquad b$`	$\boxed{a}\qquad\boxed{b}$		
`$a\hspace{0.5cm}b$`	$\boxed{a}\hspace{0.5cm}\boxed{b}$	`$a\hspace{-0.5cm}b$`	$\boxed{a}\hspace{-0.5cm}\boxed{b}$
`$a\kern0.5cm b$`	$\boxed{a}\kern0.5cm\boxed{b}$	`$a\kern-0.5cm b$`	$\boxed{a}\kern-0.5cm\boxed{b}$
`$a\hphantom{xx}b$`	$\boxed{a}\hphantom{xx}\boxed{b}$		
`$axxb$`	$\boxed{a}xx\boxed{b}$		

Table 7: Spaces in Mathmode

11.3 Vertical whitespace

11.3.1 Before/Behind math expressions

There are four predefined lenghts, which control the vertical whitespace of displayed formulas:

```
\abovedisplayskip=12pt plus 3pt minus 9pt
\abovedisplayshortskip=0pt plus 3pt
\belowdisplayskip=12pt plus 3pt minus 9pt
\belowdisplayshortskip=7pt plus 3pt minus 4pt
```

The short skips are used if the formula starts behind the end of the foregoing last line. Only for demonstration in the following examples the shortskips are set to `0pt` and the normal skips to `20pt` without any glue:

The line ends before.

$$f(x)=\int\frac{\sin x}{x}dx \tag{31}$$

The line doesn't end before the formula.

$$f(x)=\int\frac{\sin x}{x}dx \tag{32}$$

And the next line starts as usual with some text ...

```
\abovedisplayshortskip=0pt
\belowdisplayshortskip=0pt
\abovedisplayskip=20pt
\belowdisplayskip=20pt
\noindent The line ends before.
\begin{equation}
 f(x) = \int\frac{\sin x}{x}dx
\end{equation}
\noindent The line doesn't end before the formula.
\begin{equation}
 f(x) = \int\frac{\sin x}{x}dx
\end{equation}
\noindent And the next line starts as usual with some text ...
```

11.3.2 Inside math expressions

`\\[<length>]` This works inside the math mode in the same was as in the text mode.

`\jot` The vertical space between the lines for all math expressions which allow multiple lines can be changed with the length `\jot`, which is predefined as

`\newdimen\jot \jot=3pt`

The following three formulas show this for the default value, `\jot=0pt` and `\jot=10pt`.

$$\begin{aligned} y &= d \\ y &= c\frac{1}{x}+d \\ y &= b\frac{1}{x^2}+cx+d \end{aligned} \qquad \begin{aligned} y &= d \\ y &= c\frac{1}{x}+d \\ y &= b\frac{1}{x^2}+cx+d \end{aligned} \qquad \begin{aligned} y &= d \\[10pt] y &= c\frac{1}{x}+d \\[10pt] y &= b\frac{1}{x^2}+cx+d \end{aligned}$$

Defining a new environment with a parameter makes things easier, because changes to the length are locally.

```
\newenvironment{mathspace}[1]{%
  \setlength{\jot}{#1}%
  \ignorespaces%
}{%
  \ignorespacesafterend%
}
```

`\arraystretch` **\arraystretch** The vertical space between the lines for all math expressions which contain an `array` environment can be changed with the command `\arraystretch`, which is predefined as

```
\def\arraystretch{1}
```

Renewing this definition is global to all following math expressions, so it should be used in the same way than `\jot`.

12 Styles

Mode	Inline	Displayed
default	$f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\displaystyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\displaystyle`	$\displaystyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\displaystyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\scriptstyle`	$\scriptstyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\scriptstyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\scriptscriptstyle`	$\scriptscriptstyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\scriptscriptstyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\textstyle`	$\textstyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\textstyle f(t) = \frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$

Table 8: Math styles

This depends on the environment in which they are used. An inline formula has a default math fontsize called `\textstyle`, which is smaller than the one for a display formula (see section 3), which is called `\displaystyle`. Below this predefinition there are two other special fontstyles for math, `\scriptstyle` and `\scriptscriptstyle`. They are called „style“ in difference to „size“, because they have a dynamic character, their real fontsize belongs to the environment in which they are used. A fraction for example is by default in scriptstyle when it's in an inline formula like this $\frac{a}{b}$, which can be changed to $\displaystyle\frac{a}{b}$. This maybe in some cases useful but it looks in general ugly because the line spacing is too big. These four styles are predefined and together in a logical relationship. It's no problem to use the other styles like `\large`, `\Large`, ... **outside** the math environment. For example a fraction written with `\Huge`: $\Huge\frac{a}{b}$ (`\Huge$\frac{a}{b}$`). This may cause some problem when you want to write a displayed formula in another fontsize, because `\textstyle` `\displaystyle` `\scriptstyle` `\scripscriptstyle` it also affect the interline spacing of the preceding part of the paragraph. If you end the paragraph, you get problems with spacing and page breaking above the equations. So it is better to declare the font size and then restore the baselines:

$$\int_1^2 \frac{1}{x}\,dx = 0.5 \tag{33}$$

```
\makeatletter
\newenvironment{smallequation}[1]{%
  \skip@=\baselineskip
  #1%
  \baselineskip=\skip@
  \equation
}{\endequation \ignorespacesafterend}
\makeatother

\begin{smallequation}{\tiny}
\int_1^2\,\frac{1}{x}\,dx=0.5
\end{smallequation}
```

If you use this the other way round for huge fontsizes, don't forget to load package `exscale` (see section 54 on page 74).

13 Dots

\cdots
\dots
\dotsb
\dotsc
\dotsi
\dotsm
\dotso
\ldots
\vdots

In addition to the above decorations there are some more different dots which are single commands and not by default over/under a letter. It's not easy to see the differences between some of them. Dots from lower left to upper right are possible with `\reflectbox{$\ddots$}` ⋰

\cdots	$\cdots$	\ddots	$\ddots$	\dotsb	$\cdots$	\dotsc	$\ldots$	\dotsi	$\cdots$
\dotsm	$\cdots$	\dotso	$\ldots$	\ldots	$\ldots$	\vdots	$\vdots$		

Table 9: Dots in Mathmode

14 Accents

The letter „a“ is only for demonstration. The table 10 shows all in standard LaTeX available accents and the ones which are placed under a character, too. With package `amssymb` it is easy to define new accents. For more information see section 36 on page 59 or other possibilities at section 49 on page 69.

14.1 Over- and underbrackets

There are no `\underbracket` and `\overbracket` commands in the list of accents. They can be defined in the preamble with the following code.

`\acute`	$\acute{a}$	`\bar`	$\bar{a}$	`\breve`	$\breve{a}$
`\acute`	$\acute{a}$	`\bar`	$\bar{a}$	`\breve`	$\breve{a}$
`\check`	$\check{a}$	`\dddot`	$\dddot{a}$	`\ddot`	$\ddot{a}$
`\dot`	$\dot{a}$	`\grave`	$\grave{a}$	`\hat`	$\hat{a}$
`\mathring`	$\mathring{a}$	`\overbrace`	$\overbrace{a}$	`\overleftarrow`	$\overleftarrow{a}$
`\overleftrightarrow`	$\overleftrightarrow{a}$	`\overline`	$\overline{a}$	`\overrightarrow`	$\overrightarrow{a}$
`\tilde`	$\tilde{a}$	`\underbar`	$\underline{a}$	`\underbrace`	$\underbrace{a}$
`\underleftarrow`	$\underleftarrow{a}$	`\underleftrightarrow`	$\underleftrightarrow{a}$	`\underline`	$\underline{a}$
`\underrightarrow`	$\underrightarrow{a}$	`\vec`	$\vec{a}$	`\widehat`	$\widehat{a}$
`\widetilde`	$\widetilde{a}$				

Table 10: Accents in Mathmode

```
\makeatletter
\def\underbracket{%
 \@ifnextchar[{\@underbracket}{\@underbracket [\@bracketheight]}%
}
\def\@underbracket[#1]{%
 \@ifnextchar[{\@under@bracket[#1]}{\@under@bracket[#1][0.4em]}%
}
\def\@under@bracket[#1][#2]#3{%\message {Underbracket: #1,#2,#3}
 \mathop{\vtop{\m@th \ialign {##\crcr $\hfil \displaystyle {#3}\hfil $%
 \crcr \noalign {\kern 3\p@ \nointerlineskip }\upbracketfill {#1}{#2}
     \crcr \noalign {\kern 3\p@ }}}}\limits}
\def\upbracketfill#1#2{$\m@th \setbox \z@ \hbox {$\braceld$}
                 \edef\@bracketheight{\the\ht\z@}\bracketend{#1}{#2}
                 \leaders \vrule \@height #1 \@depth \z@ \hfill
                 \leaders \vrule \@height #1 \@depth \z@ \hfill \bracketend
                     {#1}{#2}$}
\def\bracketend#1#2{\vrule height #2 width #1\relax}
\makeatother
```

1. `\underbrace{...}` is an often used command:

$$\underbrace{x^2+2x+1}_{(x+1)^2} \quad = \quad f(x) \tag{34}$$

2. Sometimes an under**bracket** is needed, which can be used in more ways than `\underbrace{...}` an example for `\underbracket{...}`:

$$\text{Hate Science}\quad \underbracket{1\to2\to3\to4}_{\text{low}}\to\underbracket{5\to6\to7}_{\text{medium}}\to\underbracket{8\to9\to10}_{\text{high}}\quad\text{Love Science}$$

14.1.1 Use of `\underbracket{...}`

The `\underbracket{...}` command has two optional parameters:

- the line thickness in any valid latex unit, e.g. `1pt`
- the height of the edge brackets, e.g. `1em`

using without any parameters gives the same values for thickness and height as predefined for the `\underbrace` command.

1.	`$\underbracket {foo\ bar}$`	$\underbracket{foo\ bar}$
2.	`$\underbracket[2pt] {foo\ bar}$`	$\underbracket[2pt]{foo\ bar}$
3.	`$\underbracket[2pt] [1em] {foo\ bar}$`	$\underbracket[2pt][1em]{foo\ bar}$

14.1.2 Overbracket

In addition to the underbracket an overbracket is also useful, which can be used in more ways than `\overbrace{...}`. For example:

$$\text{Hate Science}\quad \underset{\text{low}}{\overbracket[0.5pt]{1\rightarrow 2\rightarrow 3\rightarrow 4}}\underset{\text{medium}}{\overbracket[1pt]{\rightarrow 5\rightarrow 6\rightarrow 7}}\underset{\text{high}}{\overbracket[1.5pt]{\rightarrow 8\rightarrow 9\rightarrow 10}}\quad\text{Love Science}$$

The `\overbracket{...}` command has two optional parameters:

- the line thickness in any valid latex unit, e.g. `1pt`
- the height of the edge brackets, e.g. `1em`

using without any parameters gives the same values for thickness and height as predefined for the `\overbrace` command.

1.	`$\overbracket {foo\ bar}$`	$\overbracket{foo\ bar}$
2.	`$\overbracket[2pt] {foo\ bar}$`	$\overbracket[2pt]{foo\ bar}$
3.	`$\overbracket[2pt] [1em] {foo\ bar}$`	$\overbracket[2pt][1em]{foo\ bar}$

14.2 Vectors

Especially for vectors there is the `esvect.sty`[14] package, which looks better than the `\overrightarrow`, f.ex:

Look into the documentation for more details about `esvect.sty`.

[14] CTAN://macros/latex/contrib/supported/esvect/

\vv{...}	\overrightarrow{...}
$\vec{a}$	$\overrightarrow{a}$
$\overrightarrow{abc}$	$\overrightarrow{abc}$
$\vec{\imath}$	$\overrightarrow{\imath}$
$\vec{A}_x$	$\overrightarrow{A}_x$

Table 11: Vectors with Package `esvect.sty` (in the right column the default one from LaTeX)

15 Exponents and indices

The two active characters `_` and `^` can only be used in math mode. The **following** character will be printed as an indices (`$y=a_1x+a_0$`: $y = a_1x + a_0$) or as an exponent (`$x^2+y^2=r^2$`: $x^2+y^2 = r^2$). For more than the next character put it inside of `{}`, like `$a_{i-1}+a_{i+1}<a_i$`: $a_{i-1} + a_{i+1} < a_i$.

Especially for multiple exponents there are several possiblities. For example:

$$((x^2)^3)^4 = {({(x^2)}^3)}^4 = {\left({\left(x^2\right)}^3\right)}^4 \tag{35}$$

```
((x^2)^3)^4 =
{({(x^2)}^3)}^4 =
{\left({\left(x^2\right)}^3\right)}^4
```

For variables with both, exponent and indice the order is not important, `$a_1^2$` is exactly the same than `$a^2_1$`: $a_1^2 = a^2_1$

16 Operators

They are written in upright font shape and are placed with some additional space before and behind for a better typesetting. With the `amsmath.sty` package it's possible to define one's own operators (see section 41 on page 65). Table 12 and 13 on the next page show a list of the predefined ones for standard LaTeX.

`\coprod`	$\coprod$	`\bigvee`	$\bigvee$	`\bigwedge`	$\bigwedge$
`\biguplus`	$\biguplus$	`\bigcap`	$\bigcap$	`\bigcup`	$\bigcup$
`\intop`	$\intop$	`\int`	$\int$	`\prod`	$\prod$
`\sum`	$\sum$	`\bigotimes`	$\bigotimes$	`\bigoplus`	$\bigoplus$
`\bogodot`	$\bigodot$	`\ointop`	$\oint$	`\oint`	$\oint$
`\bigsqcup`	$\bigsqcup$	`\smallint`	$\smallint$		

Table 12: The predefined operators of `fontmath.ltx`

The difference between `\intop` and `\int` is that the first one has by default over/under limits and the second subscript superscript limits. Both can be changed with the \limits or \nolimits command. The same behaviour happens to the \ointop and \oint Symbols.

\log	$\log$	\lg	$\lg$	\ln	$\ln$
\lim	$\lim$	\limsup	$\limsup$	\liminf	$\liminf$
\sin	$\sin$	\arcsin	$\arcsin$	\sinh	$\sinh$
\cos	$\cos$	\arccos	$\arccos$	\cosh	$\cosh$
\tan	$\tan$	\arctan	$\arctan$	\tanh	$\tanh$
\cot	$\cot$	\coth	$\coth$	\sec	$\sec$
\csc	$\csc$	\max	$\max$	\min	$\min$
\sup	$\sup$	\inf	$\inf$	\arg	$\arg$
\ker	$\ker$	\dim	$\dim$	\hom	$\hom$
\det	$\det$	\exp	$\exp$	\Pr	$\Pr$
\gcd	$\gcd$	\deg	$\deg$	\bmod	$\bmod$
\pmod{a}	$\pmod{a}$				

Table 13: The predefined operators of latex.ltx

For more predefined operator names see table 20 on page 69. It is easy to define a new operator with

```
\makeatletter
\newcommand\foo{\mathop{\operator@font foo}\nolimits}
\makeatother
```

Now you can use \foo in the usual way:

$$\mathop{\mathrm{foo}}\nolimits_1^2 = x^2$$

In this example \foo is defined with \nolimits, means that limits are placed in super/subscript mode and not over under. This is still possible with \limits in the definition or the equation:

$$\mathop{\mathrm{foo}}\limits_1^2 = x^2$$

```
\[ \foo\limits_1^2 = x^2 \]
```

$\mathcal{A}\!\mathcal{M}\!\mathcal{S}$math has an own macro for a definition, have a look at section 41 on page 65.

17 Greek letters

The amsmath package simulates a bold font for the greek letters, it writes a greek character twice with a small kerning. The \mathbf{<character>} doesn't work with lower greek character. See section 45 on page 68 for the \pmb macro, which makes it possible to print bold lower greek letters. Not all upper case letters have own macro names. If there is no difference to the roman font, then the default letter is used, e.g.: A for the upper case of α. The table 14 shows only those upper case letters which have own macro names. Some of the lower case letters have an additional `var` option for an alternative.

lower	default	upper	default	`\mathbf`	`\mathit`
`\alpha`	α				
`\beta`	β				
`\gamma`	γ	`\Gamma`	Γ	$\mathbf{\Gamma}$	$\mathit{\Gamma}$
`\delta`	δ	`\Delta`	Δ	$\mathbf{\Delta}$	$\mathit{\Delta}$
`\epsilon`	ϵ				
`\varepsilon`	ε				
`\zeta`	ζ				
`\eta`	η				
`\theta`	θ	`\Theta`	Θ	$\mathbf{\Theta}$	$\mathit{\Theta}$
`\vartheta`	ϑ				
`\iota`	ι				
`\kappa`	κ				
`\delta`	λ	`\Lambda`	Λ	$\mathbf{\Lambda}$	$\mathit{\Lambda}$
`\mu`	μ				
`\nu`	ν				
`\xi`	ξ	`\Xi`	Ξ	$\mathbf{\Xi}$	$\mathit{\Xi}$
`\pi`	π	`\Pi`	Π	$\mathbf{\Pi}$	$\mathit{\Pi}$
`\varpi`	ϖ				
`\rho`	ρ				
`\varrho`	ϱ				
`\sigma`	σ	`\Sigma`	Σ	$\mathbf{\Sigma}$	$\mathit{\Sigma}$
`\varsigma`	ς				
`\tau`	τ				
`\upsilon`	υ	`\Upsilon`	Υ	$\mathbf{\Upsilon}$	$\mathit{\Upsilon}$
`\phi`	ϕ	`\Phi`	Φ	$\mathbf{\Phi}$	$\mathit{\Phi}$
`\varphi`	φ				
`\chi`	χ				
`\psi`	ψ	`\Psi`	Ψ	$\mathbf{\Psi}$	$\mathit{\Psi}$
`\omega`	ω	`\Omega`	Ω	$\mathbf{\Omega}$	$\mathit{\Omega}$

Table 14: The greek letters

18 Pagebreaks

`\allowdisplaybreaks`

By default a displayed formula cannot have a pagebreak. This makes some, but sometimes it gives a better typesetting, when a pagebreak is possible.

\allowdisplaybreaks

This macro enables TEX to insert pagebreaks into displayed formulas whenever a newline command appears.

19 \stackrel

\stackrel puts a character on top of another one which may be important if a used symbol is not predefined. For example „$\stackrel{\wedge}{=}$" (\stackrel{\wedge}{=}). The syntax is

```
\stackrel{top}{base}
```

Such symbols may be often needed so that a macro definition in the preamble makes some sense:

```
\newcommand{\eqdef}{%
 \ensuremath{%
  \stackrel{\mathrm{def}}{=}%
 }%
}
```

With the \ensuremath command we can use the new \eqdef command in text- and in mathmode, LATEX switches automatically in mathmode, which saves some keystrokes like the following command, which is written without the delimiters ($...$) for the mathmode $\stackrel{\mathrm{def}}{=}$, only \eqdef with a space at the end. In mathmode together with another material it may look like $\vec{x}\stackrel{\mathrm{def}}{=}\left(x_{1},\ldots,x_{n}\right)$ and as command sequence

```
$\vec{x}\eqdef\left(x_{1},\ldots,x_{n}\right)$
```

The fontsize of the top is one size smaller than the one from the base, but it is no problem to get both in the same size, just increase the top or decrease the base.

20 \choose

\choose is like \atop with delimiters or like \frac without the fraction line and also with delimiters. It is often used for binoms and has the following syntax:

```
{above \choose below}
```

The two braces are not really important but it is safe to use them.

$$\binom{m+1}{n} = \binom{m}{n} + \binom{m}{k-1} \tag{36}$$

```
{{m+1 \choose n}}={{m \choose n}}+{{m \choose k-1}}\label{eq:choose}
```

See section 34.2 on page 58 for the $\mathcal{AMS}$math equivalents and enhancements.

21 Color in math expressions

There is no difference in using colored text and colored math expressions. With

`\usepackage{color}`

in the preamble the macro `\textcolor{<color>}{<text or math>}` exists.

$$f(x) = \int\limits_1^{\infty}\frac{1}{x}\,dx=1 \tag{37}$$

```
\begin{equation}
\textcolor{blue}{f(x)} = \int\limits_1^{\infty}\textcolor{red}{\frac
    {1}{x}}\,dx=1
\end{equation}
```

If all math expressions should be printed in the same color, then it is better to use the `everydisplay` macro (24 on page 43).

22 Boldmath

`\mathversion`
`\boldmath`
`\unboldmath`

Writing a whole formula in bold is possible with the command sequence `\boldmath ... \unboldmath`, which itself must be written in textmode (outside the formula) or with the command `{\mathversion{bold} ... }`.

$$\sum_{\substack{1\le j\le p\\1\le j\le q\\1\le k\le r}} a_{ij}b_{jk}c_{ki} \qquad \boldsymbol{\sum_{\substack{1\le j\le p\\1\le j\le q\\1\le k\le r}} a_{ij}b_{jk}c_{ki}}$$

```
\boldmath
\[
\sum_{%
 \makebox[0pt]{$%
  {{\scriptscriptstyle 1\le j\le p\atop {%
  {1\le j\le q\atop 1\le k\le r}}}}%
 $}%
 }a_{ij}b_{jk}c_{ki}
\]
\unboldmath
```

The `\mathversion` macro defines a math style which is valid for all following math expressions. If you want to have all math in bold then use this macro instead of `\boldmath`. But it is no problem to put `\mathversion` inside a group to hold the changes locally.

$$\boldsymbol{y(x) = ax^3 + bx^2 + cx + d} \tag{38}$$

```
{\mathversion{bold}%
\begin{equation}
y(x) = ax^3+bx^2+cx+d
\end{equation}}
```

Single characters inside a formula can be written in bold with `mathbf`, but only in upright mode, which is in general not useful as shown in equation 39. It is better to use package `bm.sty` (see section 48 on page 69).

$$\sum_{\substack{1\le j\le p\\1\le j\le q\\1\le k\le r}} a_{ij}\mathbf{b_{jk}}c_{ki} \tag{39}$$

22.1 Bold math expressions as part of titles and items

By default the titles in sections, subsections, a.s.o. are printed in bold. Same for the `description` environment. The problem is that a math expression in one of these environments is printed in default font shape, like the following example for a `section` and `description` environment:

> **22 Function** $f(x) = x^2$
>
> **This is** $y = f(x)$ Only a demonstration.
>
> **And** $z = f(x, y)$ Another demonstration.

With a redefinition of the `section` and `item` macros it is possible to get everything in bold font.

> **22 Function** $\boldsymbol{f(x) = x^2}$
>
> **This is** $\boldsymbol{y = f(x)}$ Only a demonstration.
>
> **And** $\boldsymbol{z = f(x, y)}$ Another demonstration.

```
\let\itemOld\item
\makeatletter
\renewcommand\item[1][]{%
    \def\@tempa{#1}
    \ifx\@tempa\@empty\itemOld\else\boldmath\itemOld[#1]\unboldmath\
        fi%
```

```
}
\makeatother
\let\sectionOld\section
\renewcommand\section[2][\empty]{%
 \boldmath\sectionOld[#1]{#2}\unboldmath%
}
```

23 Multiplying numbers

When the dot is used as the decimal marker as in the United States, the preferred sign for the multiplication of numbers or values of quantities is a cross (`\times` $\times$), not a half-high and centered dot (`\cdot` $\cdot$).

When the comma is used as the decimal marker as in Europe, the preferred sign for the multiplication of numbers is the half-high dot. The multiplication of quantity symbols (or numbers in parentheses or values of quantities in parentheses) may be indicated in one of the following ways: ab, ab, $a \cdot b$, $a \times b$.

For more information see „Nist Guide to SI Units -More on Printing and Using Symbols and Numbers in Scientific and Technical Documents“[15] or the German DIN 1304, Teil 1.

24 Other macros

`\everymath`
`\everydisplay`
`\underline`

There are some other macros which are not mentioned in the forgoing text. Here comes a not really complete list of these macros.

`\everymath` puts the argument before any inlined math expression, e.g. `\everymath{\small}`.

`\everydisplay` puts the argument before any displayed math expression, e.g. `\everydisplay{\color{blue}}`.

`\underline` underlines a math expression and has to used inside the math mode.

$$\underline{F(x) = \int f(x)\,dx}$$

[15] http://physics.nist.gov/Pubs/SP811/sec10.html

Part II
$\mathcal{AMS}$math package

In general the $\mathcal{AMS}$packages are at least a collection of three different ones:

1. `amsmath.sty`
2. `amssymb.sty`
3. `amsfonts.sty`

In the following only the first one is described in detail.
The `amsmath.sty` has the following options:

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

tbtags 'Top-or-bottom tags' 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 type - $\prod$, $\coprod$, $\bigotimes$, $\bigoplus$, and so forth - but 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 'operator names' such as det, inf, lim, 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` command -e.g., `\usepackage[intlimits]{amsmath}`. The amsmath package also recognises 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 `\usepackage{amsmath}` statement.

leqno Place equation numbers on the left.

reqno (default) Place equation numbers on the right.

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

All math environments are displayed ones, so there is no special inline math.

25 `align` environments

There are four different align environments, described in the following subsections. Their behaviour is shown in table 15. The code for all align environments was:

```
\begin{<name>}
   <name>= & x & x= & x\\
   <name>= & x & x= & x
\end{<name>}
```

25.1 The default `align` environment

The **eqnarray** environment has a not so good spacing between the cells. Writing the equations no. 2 to 5 with the **align** environment gives:

$$
\begin{aligned}
y &= d && (40)\\
y &= cx + d && (41)\\
y_{12} &= bx^2 + cx + d && (42)\\
y(x) &= ax^3 + bx^2 + cx + d && (43)
\end{aligned}
$$

The code looks like:

```
\begin{align}
 y & =d\label{eq:IntoSection}\\
 y & =cx+d\\
 y_{12} & =bx^{2}+cx+d\\
 y(x) & =ax^{3}+bx^{2}+cx+d
\end{align}
```

- The **align** environment has an implicit **{rlrl...}** horizontal alignment with a vertical column-alignment, f.ex.:

$$12 \qquad\qquad\qquad 3$$

```
\begin{align*}
  1 & 2 & 3
\end{align*}
```

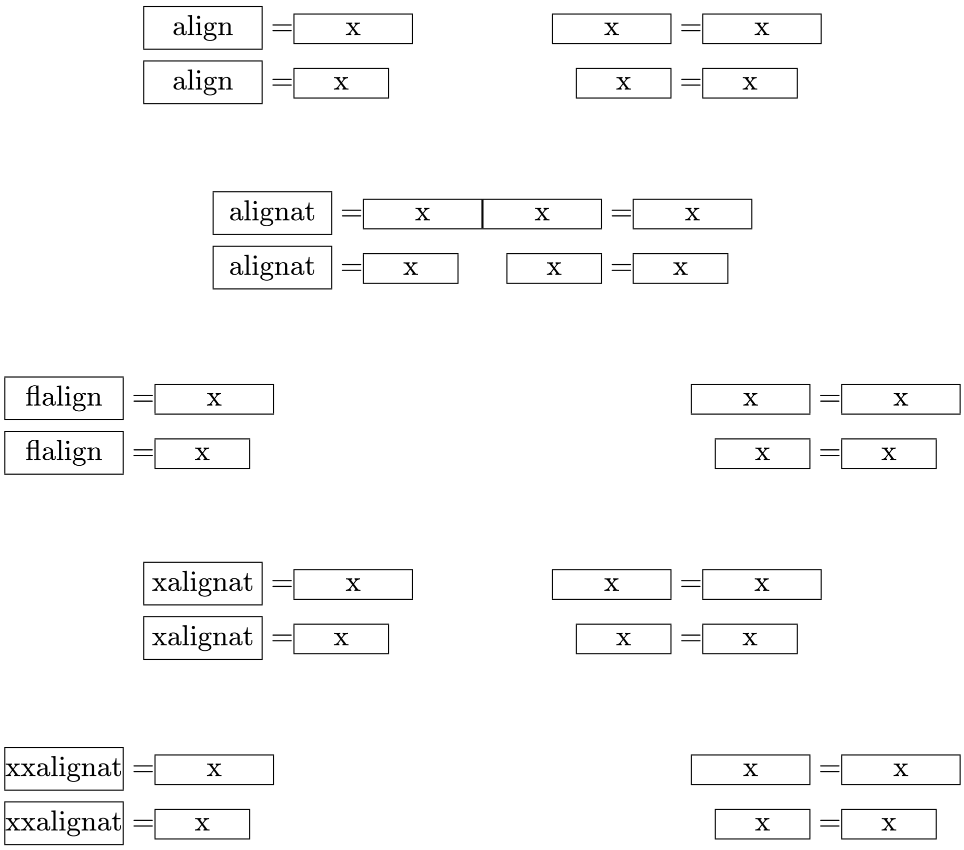


Table 15: Comparison between the different align environments with the same code, where the first three can have an equationnumber

- A nonumber-version `\begin{align*}...\end{align*}` exists.
- Not numbered single rows are possible with `\nonumber`.
- The `align` environment takes the whole horizontal space if you have more than two columns:

$$\begin{aligned} y &= d & z &= 1 && (44)\\ y &= cx+d & z &= x+1 && (45)\\ y_{12} &= bx^2+cx+d & z &= x^2+x+1 &&\\ y(x) &= ax^3+bx^2+cx+d & z &= x^3+x^2+x+1 && (46) \end{aligned}$$

The code for this example looks like

```
\begin{align}
 y & =d & z & =1\\
 y & =cx+d & z & =x+1\\
 y_{12} & =bx^{2}+cx+d & z & =x^{2}+x+1\nonumber \\
```

```
 y(x) & =ax^{3}+bx^{2}+cx+d & z & =x^{3}+x^{2}+x+1
\end{align}
```

25.2 alignat environment

> From now the counting of the equation changes. It is introduced with a foregoing command, which doesn't really make sense, it is only for demonstration:
> `\renewcommand{\theequation}{\thepart-\arabic{equation}}`.

This means „align at several places" and is something like more than two `align` environment side by side. Parameter is the number of the `align` environments, which is not important for the user. The above last `align`-example looks like:

$$y = d \qquad z = 1 \tag{II-47}$$
$$y = cx + d \qquad z = x + 1 \tag{II-48}$$
$$y_{12} = bx^2 + cx + d \qquad z = x^2 + x + 1$$
$$y(x) = ax^3 + bx^2 + cx + d \qquad z = x^3 + x^2 + x + 1 \tag{II-49}$$

The parameter was 2 and is for the following example 3:

$$i_{11} = 0.25 \qquad i_{12} = i_{21} \qquad i_{13} = i_{23}$$
$$i_{21} = \frac{1}{3} i_{11} \qquad i_{22} = 0.5 i_{12} \qquad i_{23} = i_{31} \tag{II-50}$$
$$i_{31} = 0.33 i_{22} \qquad i_{32} = 0.15 i_{32} \qquad i_{33} = i_{11} \tag{II-51}$$

For this example the code is:

```
\begin{alignat}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22}\quad & i_{32} & =0.15i_{32}\quad & i_{33} & =i_{11}
\end{alignat}
```

- The **`alignat`** environment has an implicit {rlrl...rlrl} horizontal alignment with a vertical column-alignment.
- A nonumber-version `\begin{alignat*}...\end{alignat*}` exists.
- Not numbered single rows are possible with `\nonumber`.

25.3 flalign environment

This is the new replacement for the xalignat and xxalignat environments. It is nearly the same as the xalignat environment, only a little more „out spaced".

$$\begin{aligned} 3i_{11} &= 0.25 & i_{12} &= i_{21} & i_{13} &= i_{23} \\ i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31} \quad \text{(II-52)} \\ i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11} \quad \text{(II-53)} \end{aligned}$$

The same code looks like:

```
\begin{flalign}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22}\quad & i_{32} & =0.15i_{32}\quad & i_{33} & =i_{11}
\end{flalign}
```

This environment can be used to mix centered and left aligned equations without using the document wide valid option fleqn.

$$f(x) = \int \frac{1}{x}\,dx \tag{II-54}$$

$$f(x) = \int \frac{1}{x}\,dx \tag{II-55}$$

Equation II-55 is left aligned in fact of the second tabbing character &.

```
\begin{align}\label{eq:centered}
 f(x) & = \int\frac{1}{x}\,dx
\end{align}

\begin{flalign}\label{eq:leftaligned}
 f(x) & = \int\frac{1}{x}\,dx &
\end{flalign}
```

25.4 xalignat environment

This is an obsoletee macro but still supported by the amsmath package. Same as alignat environment, only a little more „out spaced".

$$\begin{aligned} i_{11} &= 0.25 & i_{12} &= i_{21} & i_{13} &= i_{23} \\ i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31} \quad \text{(II-56)} \\ i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11} \quad \text{(II-57)} \end{aligned}$$

The same code looks like:

```
\begin{xalignat}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22}\quad & i_{32} & =0.15i_{32}\quad & i_{33} & =i_{11}
\end{xalignat}
```

25.5 `xxalignat` environment

Like **`xalignat`** an obsoletee macro but still supported by the **`amsmath`** package. Same as **`align`** environment, only extremely „out spaced“, therefore no equation number!

$$
\begin{aligned}
i_{11} &= 0.25 & i_{12} &= i_{21} & i_{13} &= i_{23}\\
i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31}\\
i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11}
\end{aligned}
$$

The same code looks like:

```
\begin{xxalignat}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22} & i_{32} & =0.15i_{32} & i_{33} & =i_{11}
\end{xxalignat}
```

25.6 `aligned` environment

`\begin{aligned}`
`...`
`\end{aligned}`

In difference to the **`split`** environment (section 28 on page 52), the **`aligned`** environment allows more than one horizontal alignment and has only one equation number:

$$
\begin{aligned}
2x+3 &= 7 & 2x+3-3 &= 7-3\\
2x &= 4 & \frac{2x}{2} &= \frac{4}{2}\\
x &= 2
\end{aligned} \tag{II-58}
$$

```
\begin{equation}
\begin{aligned}
    2x+3 &= 7 &     2x+3-3 &= 7-3   \\
    2x   &= 4  & \frac{2x}2 &= \frac42\\
    x    &= 2
\end{aligned}
\end{equation}
```

The **aligned** environment is similiar to the **array** environment, there exists no starred version and it has only one equation number and has to be part of another math environment, which should be **equation** environment. The advantage of **aligned** is the much more better horizontal and vertical spacing.

26 gather environment

This is like a multi line environment with no special horizontal alignment. All rows are centered and can have an own equation number:

$$i_{11} = 0.25 \tag{II-59}$$
$$i_{21} = \frac{1}{3}i_{11}$$
$$i_{31} = 0.33i_{22} \tag{II-60}$$

For this example the code looks like:

```
\begin{gather}
  i_{11} = 0.25\\
  i_{21} = \frac{1}{3}i_{11}\nonumber\\
  i_{31} =0.33i_{22}
\end{gather}
```

- The **gather** environment has an implicit `{c}` horizontal alignment with no vertical column-alignment. It's just like an one-column array/table.
- A nonumber-version `\begin{gather*}...\end{gather*}` exists. Look at section 28 on page 52 for an example.

27 multline environment

This is also like a multi line[16] environment with a special vertical alignment. The **first** row is **left aligned**, the second and all following ones except the last one are **centered** and the **last** line is **right aligned**. It's often used to write extremely long formulas:

```
\begin{multline}
  A = \lim _{n\rightarrow \infty }\Delta x\left( a^{2}+\left( a^{2}+2a\Delta x
        +\left( \Delta x\right) ^{2}\right)\right.\\
  +\left( a^{2}+2\cdot 2a\Delta x+2^{2}\left( \Delta x\right) ^{2}\right)\\
  +\left( a^{2}+2\cdot 3a\Delta x+3^{2}\left( \Delta x\right) ^{2}\right)\\
  + \ldots\\
```

[16] It's no typo, the name of the environment is **multline**, no missing i here!

$$\begin{multline}
\framebox[0.5\linewidth]{x}\\
\framebox[0.5\linewidth]{x}\\
\framebox[0.5\linewidth]{x}\\
\shoveright{\framebox[0.5\linewidth]{x}}\\
\framebox[0.5\linewidth]{x}\\
\framebox[0.5\linewidth]{x}
\tag{II-62}
\end{multline}$$

Figure 1: `multline` Alignment Demo (the fourth row is shifted to the right with `\shoveright`)

```
  \left.+\left( a^{2}+2\cdot (n-1)a\Delta x +(n-1)^{2}\left( \Delta x\right) ^{2}\
    right) \right)\\
  = \frac{1}{3}\left( b^{3}-a^{3}\right)
\end{multline}
```

$$\begin{multline}
A=\lim_{n\to\infty}\Delta x\left(a^{2}+\left(a^{2}+2a\Delta x+\left(\Delta x\right)^{2}\right)\right.\\
+\left(a^{2}+2\cdot 2a\Delta x+2^{2}\left(\Delta x\right)^{2}\right)\\
+\left(a^{2}+2\cdot 3a\Delta x+3^{2}\left(\Delta x\right)^{2}\right)\\
+\ldots\\
\left.+\left(a^{2}+2\cdot(n-1)a\Delta x+(n-1)^{2}\left(\Delta x\right)^{2}\right)\right)\\
=\frac{1}{3}\left(b^{3}-a^{3}\right)
\tag{II-61}
\end{multline}$$

- A nonumber-version `\begin{multline*}...\end{multline*}` exists.
- By default only the last line (for right equation numbers) or the first line (for left equation numbers) gets a number, the others can't.
- The alignment of a single line can be changed with the command `\shoveright` (figure 1)
- The first line and the last line have a small gap to the text border[17]. See figure 2, where the length of `\multlinegap` is set to `0pt` for the right one.

[17] When the first (numbers left) or last line (numbers right) has an equationnumber then `\multlinegap` is not used for these ones, only for the line without a number.

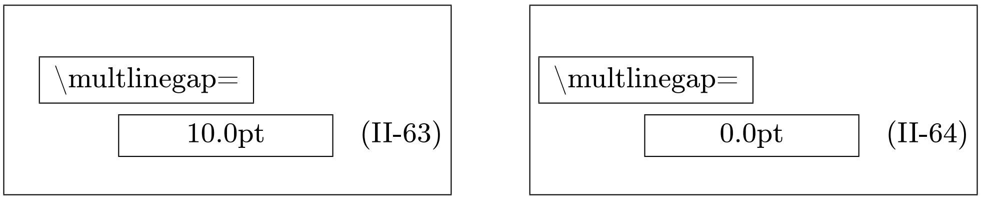


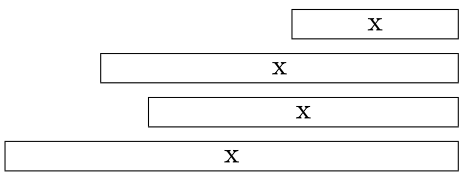
Figure 2: Demonstration of `\multlinegap` (default - 0pt)

28 `split` environment

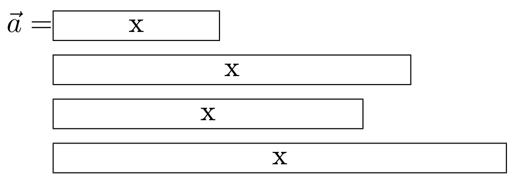
From now the counting of the equation changes. It is introduced with a foregoing command, which doesn't really make sense, it is only for demonstration:

```
\makeatletter
\@removefromreset{equation}{section}
\makeatother
```

The `split` environment is like the `multline`- or `array` environment for equations longer than the column width. Just like the array environment and in contrast to `multline`, `split` can only be used as **part of another environment**. `split` itself has no own numbering, this is given by the other environment. Without an ampersand all lines in the split environment are right-aligned and can be aligned at a special point by using an ampersand. In difference to the `aligned` environment (section 25.6 on page 49), the `split` environment permits more than one horizontal alignment.



```
\[
\begin{split}
   \framebox[0.35\columnwidth]{x}\\
   \framebox[0.75\columnwidth]{x}\\
   \framebox[0.65\columnwidth]{x}\\
   \framebox[0.95\columnwidth]{x}
\end{split}
\]
```



```
\[
\begin{split}
   \vec{a} = &\framebox[0.35\columnwidth]{x}\\
             &\framebox[0.75\columnwidth]{x}\\
             &\framebox[0.65\columnwidth]{x}\\
             &\framebox[0.95\columnwidth]{x}
\end{split}
\]
```

The following example shows the `split` environment as part of the `equation` environment:

$$\begin{split} A_{1} & = \left| \int _{0}^{1}(f(x)-g(x))dx\right| +\left| \int _{1}^{2}(g(x)-h(x))dx\right| \\ & = \left| \int _{0}^{1}(x^{2}-3x)dx\right| +\left| \int _{1}^{2}(x^{2}-5x+6)dx\right| \\ & = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right| _{0}^{1}+ \left| \frac{x^{3}}{3}-\frac{5}{2}x^{2}+6x\right| _{1}^{2}\\ & = \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-\left(\frac{1}{3}-\frac{5}{2}+6\right) \right| \\ & = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right| =\frac{7}{6}+\frac{5}{6}=2\, \textrm{FE} \end{split} \tag{II-65}$$

```
\begin{equation}
  \begin{split}
    A_{1} & = \left| \int _{0}^{1}(f(x)-g(x))dx\right| +\left|
   \int _{1}^{2}(g(x)-h(x))dx\right| \\
          & = \left| \int _{0}^{1}(x^{2}-3x)dx\right| +\left|
   \int _{1}^{2}(x^{2}-5x+6)dx\right| \\
          & = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right| _{0}^{1}+
   \left| \frac{x^{3}}{3}-
               \frac{5}{2}x^{2}+6x\right| _{1}^{2}\\
          & = \left| \frac{1}{3}-\frac{3}{2}\right| +\left|
   \frac{8}{3}-\frac{20}{2}+12-
               \left( \frac{1}{3}-\frac{5}{2}+6\right) \right| \\
          & = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}
   \right| =\frac{7}{6}+\frac{5}{6}=2\, \textrm{FE}
  \end{split}
\end{equation}
```

The same using the `array` environment with `{rl}`-alignment instead of `split` gives same horizontal alignment but another vertical spacing[18] and the symbols only in scriptsize and not textsize:[19]

$$\begin{array}{rl} A_{1} & = \left| \int _{0}^{1}(f(x)-g(x))dx\right| +\left| \int _{1}^{2}(g(x)-h(x))dx\right| \\ & = \left| \int _{0}^{1}(x^{2}-3x)dx\right| +\left| \int _{1}^{2}(x^{2}-5x+6)dx\right| \\ & = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right| _{0}^{1}+ \left| \frac{x^{3}}{3}-\frac{5}{2}x^{2}+6x\right| _{1}^{2}\\ & = \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-\left(\frac{1}{3}-\frac{5}{2}+6\right) \right| \\ & = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right| =\frac{7}{6}+\frac{5}{6}=2\, \textrm{FE} \end{array} \tag{II-66}$$

- There exists no star version (`\begin{split*}`) of the `split` environment.

[18] can be changed with \

[19] see section 12 on page 33

29 Specials for multline and split Environments

With the multline environment the equation 28 on page 26 looks like:

$$\begin{multline}\frac{1}{2}\Delta(f_{ij}f^{ij})=2\left(\sum_{i<j}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\right.\\ \left.+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}\left[2\nabla_{i}R_{jk}-\nabla_{k}R_{ij}\right]\right) \tag{II-67}\end{multline}$$

which is again a bad typesetting because of the two unequal parentheses. Each one has a size which is correct for the line but not for the whole formula. LaTeX accepts only pairs of parentheses for one line and has an „empty“ parentheses, the dot „`\left.`“ or „`\right.`“to get only one of the „pair“. There are different solutions to get the right size of the parentheses. One of them is to use the `\vphantom` command, which reserves the vertical space without any horizontal one, like a vertical rule without any thickness. The sum-symbol from the first line is the biggest one and responsible for the height, so this one is the argument of `\vphantom` which has to be placed anywhere.

$$\begin{multline}\frac{1}{2}\Delta(f_{ij}f^{ij})=2\left(\sum_{i<j}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\right.\\ \left.+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}\left[2\nabla_{i}R_{jk}-\nabla_{k}R_{ij}\right]\vphantom{\sum_{i<j}}\right) \tag{II-68}\end{multline}$$

```
\begin{multline}
\frac{1}{2}\Delta(f_{ij}f^{ij})=
 2\left(\sum_{i<j}\chi_{ij}(\sigma_{i}-
 \sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\right.\\
 \left.+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+
 f^{ij}f^{k}\left[2\nabla_{i}R_{jk}-
 \nabla_{k}R_{ij}\right]\vphantom{\sum_{i<j}}\right)
\end{multline}
```

Instead of using the `\vphantom` command it's also possible to use fixed-width parentheses, which is described in section 8 on page 24.

30 cases environment

This gives support for an often used mathematical construct.

You can also choose the more than once described way to convert some text into math, like

```
$x=\begin{cases}
 0 & \text{if A=...}\\
 1 & \text{if B=...}\\
 x & \textrm{this runs with as much text as you like,
      but without an automatic linebreak, it runs out
      of page....}
\end{cases}$
```

which gives equation II-69. It's obvious what's the problem is.

$$x = \begin{cases} 0 & \text{if A=...} \\ 1 & \text{if B=...} \\ x & \text{this runs with as much text as you like, but without a linebreak, it runs out of page....} \end{cases} \tag{II-69}$$

In this case it's better to use a parbox for the text part with a `flushleft` command for a better view.

$$x = \begin{cases} 0 & \text{if A=...} \\ 1 & \text{if B=...} \\ x & \parbox{5cm}{\flushleft this runs with as much text as you like, but without an automatic linebreak, it runs out of page....} \end{cases} \tag{II-70}$$

```
\begin{equation}
x=\begin{cases}
  0 & \text{if A=...}\\
  1 & \text{if B=...}\\
  x & \parbox{5cm}{%
      \flushleft%
      this runs with as much text as you like,
      but without an automatic linebreak,
      it runs out of page....}%
\end{cases}
\end{equation}
```

From now the counting of the equation changes. It is introduced with a foregoing command, which doesn't really make sense, it is only for demonstration:

```
\renewcommand\theequation{\arabic{equation}}
```

31 Matrix Environments

All matrix environments can be nested and an element may also contain any other math environment, so that very complex structures are possible.

\Vmatrix	$\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$	\Bmatrix	$\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$	\matrix	$\begin{matrix} a & b \\ c & d \end{matrix}$
\vmatrix	$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$	\bmatrix	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	\pmatrix	$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Table 16: Matrix environments

By default all cells have a centered alignment, which is often not the best when having different decimal numbers or plus/minus values. Changing the alignment to right is possible with

```
\makeatletter
\def\env@matrix{\hskip -\arraycolsep
  \let\@ifnextchar\new@ifnextchar
  \array{*\c@MaxMatrixCols r}}
\makeatother
```

For dots over several columns look for `\hdotsfor` in the following section.

32 Vertical Whitespace

See section 11.3 on page 31 for the lengthes which control the vertical whitespace. There is no difference to `amsmath`.

33 Dots

In addition to the section 13 on page 34 amsmath has two more commands for dots: `\dddot`{...} and `\ddddot`{...}

`\dddot{y}`|: $\dddot{y}$

`$\ddddot{y}$`: $\ddddot{y}$

Another interesting dot command is `\hdotsfor` with the syntax:

```
\hdotsfor[<spacing factor>]{<number of columns>}
```

With the spacing factor the width of the dots can be stretched or shrinked. The number of columns allows a continuing dotted line over more columns.

Equation 71 shows the definition of a tridiagonal matrix.

$$\underline{A}=\left[\begin{array}{ccccccc}
a_{11} & a_{12} & 0 & \ldots & \ldots & \ldots & 0\\
a_{21} & a_{22} & a_{23} & 0 & \ldots & \ldots & 0\\
0 & a_{32} & a_{33} & a_{34} & 0 & \ldots & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
\hdotsfor{7}\cr\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & \ldots & 0 & a_{n-2,n-3} & a_{n-2,n-2} & a_{n-2,n-1} & 0\\
0 & \ldots & \ldots & 0 & q_{n-1,n-2} & a_{n-1,n-1} & a_{n-1,n}\\
0 & \ldots & \ldots & \ldots & 0 & a_{n,n-1} & a_{nn}
\end{array}\right] \tag{71}$$

```
\begin{equation}
\underline{A}=\left[\begin{array}{ccccccc}
a_{11} & a_{12} & 0 & \ldots & \ldots & \ldots & 0\\
a_{21} & a_{22} & a_{23} & 0 & \ldots & \ldots & 0\\
0 & a_{32} & a_{33} & a_{34} & 0 & \ldots & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
\hdotsfor{7}\cr\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & \ldots & 0 & a_{n-2,n-3} & a_{n-2,n-2} & a_{n-2,n-1} & 0\\
0 & \ldots & \ldots & 0 & q_{n-1,n-2} & a_{n-1,n-1} & a_{n-1,n}\\
0 & \ldots & \ldots & \ldots & 0 & a_{n,n-1} & a_{nn}
\end{array}\right]
\end{equation}
```

34 fraction commands

34.1 Standard

Additional to the font size problem described in subsection 2.2 on page 8 `amsmath.sty` supports some more commands for fractions. The `\frac` command described in [3], does no more exist in `amsmath.sty`.

- The global fraction definition has five parameters

  ```
  \genfrac{<left delim>}{<right delim>}{<thickness>}{<mathstyle
      >}{<nominator>}{<denominator>}
  ```

  where thickness can have any length with a valid unit like `genfrac{}{}{1pt}{}{x^2+x+1}{3x-2}` $\longrightarrow \frac{x^2+x+1}{3x-2}$

- `\cfrac` (continued fraction) which is by default set in the display mathstyle and useful for fractions like

$$\cfrac{1}{\sqrt{2}+\cfrac{1}{\sqrt{3}+\cfrac{1}{\sqrt{4}+\cfrac{1}{\dots}}}} \tag{72}$$

which looks with the default `\frac` command like

$$\frac{1}{\sqrt{2}+\frac{1}{\sqrt{3}+\frac{1}{\sqrt{4}+\frac{1}{\dots}}}} \tag{73}$$

where the mathstyle decreases for every new level in the fraction. The `\cfrac` command can be called with an optional parameter which defines the placing of the nominator, which can be [l]eft, [r]ight or [c]enter (the default - see equ. 72):

$$\cfrac[l]{1}{\sqrt{2}+\cfrac[l]{1}{\sqrt{3}+\cfrac[l]{1}{\sqrt{4}+\cfrac[l]{1}{\dots}}}} \qquad \cfrac[r]{1}{\sqrt{2}+\cfrac[r]{1}{\sqrt{3}+\cfrac[r]{1}{\sqrt{4}+\cfrac[r]{1}{\dots}}}}$$

- `\dfrac` which takes by default the displaystyle, so that fractions in inline mode $\dfrac{1}{2}$ have the same size than in display mode.
- `\tfrac` (vice versa to `\dfrac`) which takes by default the scriptstyle, so that fractions in display mode have the same size than in inline mode.

$\tfrac{2}{3}$ `\tfrac{2}{3}`

$\frac{2}{3}$ `\frac{2}{3}`

34.2 Binoms

`\binom`
`\dbinom`
`\tbinom`

They are like fractions without a rule and its syntax is different to the `\choose` command from standard LaTeX (see section 2.2 on page 8). Ams-math provides three different commands for binoms just like the ones for fractions.

Command	Inlinemath	Displaymath
`\binom{m}{n}`	$\binom{m}{n}$	$\dbinom{m}{n}$
`\dbinom{m}{n}`	$\dbinom{m}{n}$	$\dbinom{m}{n}$
`\tbinom{m}{n}`	$\tbinom{m}{n}$	$\tbinom{m}{n}$

Table 17: binom commands

35 Roots

\leftroot
\uproot

The typesetting for roots is sometimes not the best. Some solutions for better typesetting are described in section 7 on page 23 for standard LaTeX. **amsmath.sty** has some more commands for the n-th root:

```
\leftroot{<number>}
\uproot{<number>}
```

<number> indicates a value for the points[20] of which the root can be adjusted to the left and to the top.

35.1 Roots with `\smash` Command

`\smash`

The default for a root with λ_{k_i} as root argument looks like $\sqrt{\lambda_{k_i}}$, which maybe not the best typesetting. It's possible to reduce the lowest point of the root to the baseline with the `\smash` command: $\sqrt{\lambda_{k_i}} \xrightarrow{\text{with \textbackslash smash}} \sqrt{\smash[b]{\lambda_{k_i}}}$

The syntax of the with the package `amsmath.sty` renewed `\smash` command[21] is

```
\smash[<position>]{<argument>}
```

The optional argument for the position can be:

t keeps the bottom and annihilates the top

b keeps the top and annihilates the bottom

tb annihilates top and bottom (the default)

36 Accents

With the macro `\mathaccent` it is easy to define new accent types, for example

[20] in PostScript units (bp - pixel).
[21] In `latex.ltx` `\smash` is defined without an optional argument.

```
\def\dotcup{$\mathaccent\cdot\cup$}
```

$\dot{\cup}$

Overwriting of two symbols is also possible:

$\curvearrowleft\curvearrowright\curvearrowleft\curvearrowright$

In this case the second symbol has to be shifted to left for a length of $5mu$ (mu: math unit).

```
\def\curvearrowleftright{%
 \ensuremath{%
  \mathaccent\curvearrowright{\mkern-5mu\curvearrowleft}%
 }%
}
```

For other possibilities to define new accent see section 49 on page 69.

37 \mod command

The modulo command is in standard LaTeX not an operator, though it's often used in formulas. amsmath.sty provides two (three) different commands for modulo, which are listed in tabular 18.

- They all insert some useful space before and behind the mod-operator.

a\mod{n^2}=b	$\rightarrow$	$a \mod n^2 = b$
a\pmod{n^2}=b	$\rightarrow$	$a \pmod{n^2} = b$
a\pod{n^2}=b	$\rightarrow$	$a \pod{n^2} = b$

Table 18: The modulo commands and their meaning

38 Equation numbering

See section 3.3 on page 13 for equation numbering. It's mostly the same, only one command is new to **amsmath.sty**. If you want a numbering like „40" then write in the preamble or like this example anywhere in your doc: \numberwithin

```
\numberwithin{equation}{section}
```

From now the numbering looks like equation 40 on page 45. For the book-class you can get the same for chapters.

If you want to get rid of the parentheses then write in preamble:

```
\makeatletter
\def\tagform@#1{\maketag@@@{\ignorespaces#1\unskip\@@italiccorr}}
\makeatother
```

Now the following four subequation numbers have no parentheses.

38.1 Subequations

Amsmath supports this with the environment **subequation**. For example:

$$y = d \tag*{38.74a}$$
$$y = cx + d \tag*{38.74b}$$
$$y = bx^2 + cx + d \tag*{38.74c}$$
$$y = ax^3 + bx^2 + cx + d \tag*{38.74d}$$

```
\begin{subequations}
\begin{align}
y & = d\\
y & = cx+d\\
y & = bx^{2}+cx+d\\
y & = ax^{3}+bx^{2}+cx+d
\end{align}
\end{subequations}
```

Inside of subequations only complete other environments (`\begin{...}` ... `\end{...}`) are possible.

```
\renewcommand{\theequation}{%
  \theparentequation{}-\arabic{equation}%
}
```

$$y = d \tag{38.75-1}$$
$$y = cx + d \tag{38.75-2}$$
$$y = bx^2 + cx + d \tag{38.75-3}$$
$$y = ax^3 + bx^2 + cx + d \tag{38.75-4}$$

A ref to a subequation is possible like the one to equation 38.75-2. The environment chooses the same counter „**equation**“ but saves the old value into „**parentequation**“.

It is also possible to place two equations side by side with counting as subfigures:

$$y = f(x) \qquad (38.76a) \qquad\qquad y = f(z) \qquad (38.76b)$$

In this case, the amsmath internal subfigure counter cannot be used and an own counter has to be defined:

```
\newcounter{mySubCounter}
\newcommand{\twocoleqn}[2]{
    \setcounter{mySubCounter}{0}%
    \let\OldTheEquation\theequation%
    \renewcommand{\theequation}{\OldTheEquation\alph{mySubCounter}}%
    \noindent%
    \begin{minipage}{.49\textwidth}
          \begin{equation}\refstepcounter{mySubCounter}
            #1
        \end{equation}
    \end{minipage}\hfill%
    \addtocounter{equation}{-1}%
    \begin{minipage}{.49\textwidth}
        \begin{equation}\refstepcounter{mySubCounter}
            #2
        \end{equation}
    \end{minipage}%
    \let\theequation\OldTheEquation
}
[ ... ]
\twocoleqn{y=f(x)}{y=f(z)}
```

39 Labels and Tags

For the `\label` command see section 3.4 on page 15, it's just the same behaviour. `amsmath.sty` allows to define own single „equation numbers" with the `\tag` command. \tag

$$f(x) = a \tag{linear}$$
$$g(x) = dx^2 + cx + b \tag{quadratic}$$
$$h(x) = \sin x \tag*{trigonometric}$$

```
\begin{align}
f(x) & =a\tag{linear}\label{eq:linear}\\
g(x) & =dx^{2}+cx+b\tag{quadratic}\label{eq:quadratic}\\
h(x) & =\sin x\tag*{trigonometric}
\end{align}
```

- The `\tag` command is also possible for unnumbered equations, LaTeX changes the behaviour when a tag is detected.
- There exists a star version `\tag{*}{...}`, which supresses any annotations like parentheses for equation numbers.
- There exists two package-options for tags, `ctagsplit` and `righttag` (look at the beginning of this part on page 44).

40 Limits

By default the sum/prod has the limits above/below and the integral at the side. To get the same behaviour for all symbols which can have limits load the packags amsmath in the preamble as

```
\usepackage[sumlimits,intlimits]{amsmath}
```

There exists also options for the vice versa (see page 44). See also section ?? for the additional commands `\underset` and `\overset`.

40.1 Multiple Limits

For general information about limits read section 2.1 on page 7. Standard LaTeX provides the `\atop` command for multiple limits (section 6 on page 22). `amsmath` has an additional command for that, which can have several lines with the following syntax:

\substack
`\begin{Sb}`
...
`\end{Sb}`
`\begin{Sp}`
...
`\end{Sp}`

```
\substack{...\\...\\...}
```

The environments described in [3]

```
\begin{Sb} ... \end{Sb}
\begin{Sp} ... \end{Sp}
```

are obsolete and no more part of `amsmath.sty`.

The example equation 21 on page 22 with the `\substack` command looks like:

$$\sum_{\substack{1\le i\le p\\1\le j\le q\\1\le k\le r}} a_{ij}b_{jk}c_{ki} \tag{40.1}$$

Insert these limits in the following way:

```
\begin{equation}
 \sum_{%
  \substack{1\le i\le p\\
   1\le j\le q\\
   1\le k\le r}
  }%
 a_{ij}b_{jk}c_{ki}
\end{equation}
```

40.2 Problems

There are still some problems with limits and the following math expression. For example:

$$X = \sum_{1\le i\le j\le n} X_{ij}$$

```
\[
X = \sum_{1\le i\le j\le n}X_{ij}
\]
```

does not look nice because of the long limit. Using a `\makebox` also does not really solve the problem, because `\makebox` is in TEX horizontal mode and knows nothing about the appropriate math font size, because limits have a smaller font size. It is better to define a `\mathclap` macro, similiar to the two macros `\llap` and `\rlap` and uses the also new defined `\clap` macro:

```
\def\clap#1{\hbox to 0pt{\hss#1\hss}}
\def\mathclap{\mathpalette\mathclapinternal}
\def\mathclapinternal#1#2{%
 \clap{$\mathsurround=0pt#1{#2}$}%
}
```

Now we can write limits which have a boxwidth of 0pt and the right font size and the following math expression appears just behind the symbol:

$$X = \sum_{\mathclap{1\le i\le j\le n}} X_{ij}$$

```
\[
X = \sum_{\mathclap{1\le i\le j\le n}}X_{ij}
\]
```

40.3 \sideset

This is a command for a very special purpose, to combine over/under limits with super/subscripts for the sum-symbol. For example: it is not possible `\sideset`
to place the prime for the equation 40.2 near to the sum-symbol, because it becomes an upper limit when writing without an preceeding `{}`.

$$\sum_{\substack{n<k\\ n\ \text{odd}}}{}' nE_n \tag{40.2}$$

The command `\sideset` has the syntax

```
\sideset{<before>}{<behind>}
```

It can place characters on all four corners of the sum-symbol:

$$\sideset{_{LowerLeft}^{UpperLeft}}{_{LowerRight}^{UpperRight}}\sum_{B}^{T}$$

```
\[
\sideset{_{LowerLeft}^{UpperLeft}}{_{LowerRight}^{UpperRight}}\sum_{
    B}^{T}
\]
```

Now it is possible to write the equation 40.2 in a proper way with the command `\sideset{}{'}` before the sum symbol:

$$\sideset{}{'}\sum_{\substack{n<k\\ n\ \mathrm{odd}}} nE_n \tag{40.3}$$

41 Operator Names

\operatorname

By default variables are written in italic and operator names in upright mode, like $y = \sin(x)$.[22] This happens only for the known operator names, but creating a new one is very easy with:

```
\newcommand{\mysin}{\operatorname{mysin}}
```

Now `\mysin` is also written in upright mode $y = \operatorname{mysin}(x)$ and with some additional space before and behind.

It's obvious, that only those names can be defined as new operator names which are not commands in another way. Instead of using the new definition as an operator, it's also possible to use the text mode. But it's better to have all operators of the same type, so that changing the style will have an effect for all operators.

\operatornamewithlimi

The new defined operator names can't have limits, only super/subscript is possible. `amsopn.sty` has an additional command `\operatornamewithlimits`, which supports over/under limits like the one from `\int` or `\sum`.

\mathop

It is also possible to use the macro `\mathop` to declare anything as operator, like

$$\sideset{_1}{}{\mathop{\mathrm{B}}}$$

```
\[ \sideset{_1}{}{\mathop{\mathrm{B}}} \]
```

With this definition it is possible to use `\sideset` for a forgoing index, which is only possible for an operator.

For a real LaTeX definition have a look at section 16 on page 37.

42 Text in Mathmode

If you need complex structures between formulas, look also at section 65.

[22] See section 16 on page 37, where all for standard LaTeX known operator names are listed. Package `amsmath` has some more (see documentation).

42.1 \text command

This is the equivalent command to `\mathrm` or `\mbox` from the standard LaTeX.

For example: $\boxed{f(x)=x\quad\text{this was math}}$.

```
$\boxed{f(x)=x\quad\text{this was math}}$
```

42.2 \intertext command

This is useful when you want to place some text between two parts of math stuff without leaving the mathmode, like the name „intertext“ says. For example we write the equation II-65 on page 53 with an additional command after the second line.

$$\begin{aligned} A_{1} &= \left| \int_{0}^{1}(f(x)-g(x))dx\right| +\left| \int_{1}^{2}(g(x)-h(x))dx\right| \\ &= \left| \int_{0}^{1}(x^{2}-3x)dx\right| +\left| \int_{1}^{2}(x^{2}-5x+6)dx\right| \end{aligned}$$

Now the limits of the integrals are used

$$\begin{aligned} &= \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right|_{0}^{1}+\left| \frac{x^{3}}{3}-\frac{5}{2}x^{2}+6x\right|_{1}^{2} \\ &= \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-\left(\frac{1}{3}-\frac{5}{2}+6\right) \right| \\ &= \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right| = \frac{7}{6}+\frac{5}{6} = 2\,\mathrm{FE} \end{aligned}$$

The code looks like:

```
\begin{equation}
  \begin{split}
    A_{1} & = \left| \int _{0}^{1}(f(x)-g(x))dx\right| +\left| \int _{1}^{2}(g(x)-h(x))dx\right| \\
          & = \left| \int _{0}^{1}(x^{2}-3x)dx\right| +\left| \int _{1}^{2}(x^{2}-5x+6)dx\right| \\
          \intertext{Now the limits of the integrals are used}
          & = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right| _{0}^{1}+\left| \frac{x^{3}}{3}-
            \frac{5}{2}x^{2}+6x\right| _{1}^{2}\\
          & = \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-
            \left( \frac{1}{3}-\frac{5}{2}+6\right) \right| \\
```

```
        & = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right
            | =\frac{7}{6}+
              \frac{5}{6}=2\, \textrm{FE}
  \end{split}
\end{equation}
```

43 Extensible Arrows

\xrightarrow

To write something like $\xrightarrow[\text{below}]{\text{above the arrow}}$ you can use the following macro

`$\xrightarrow[\text{below}]{\text{above the arrow}}$`

and the same with `\xleftarrow`. You can define your own extensible arrow macros if you need other than these two predefined ones. To get a doublelined extensible arrow like `$\Longleftrightarrow$` ($\Longleftrightarrow$) but with the same behaviour than an extensible one, write in preamble

```
\newcommand{\xLongLeftRightArrow}[2][]{%
    \ext@arrow 0055{\LongLeftRightArrowfill@}{#1}{#2}%
}
\def\LongLeftRightArrowfill@{%
 \arrowfill@\Leftarrow\Relbar\Rightarrow%
}
```

The three parts `\Leftarrow\Relbar\Rightarrow` define left|middle|right of the arrow, where the middle part would be stretched in a way that the arrow is at least as long as the text above and/or below it. This macro has one optional and one standard parameter. The optional one is written below and the standard above this arrow. Now we can write

`$\xLongLeftRightArrow[\text{below}]{\text{above the arrow}}$`

to get $\xLongLeftRightArrow[\text{below}]{\text{above the arrow}}$.

44 Frames

\boxed

`amsmath` knows the macro `\boxed` which can be used for inline $a\boxed{b+c}$ and displayed math expressions:

$$\boxed{f(x)=\int_1^\infty \frac{1}{x}\,dt=1} \tag{44.1}$$

```
\begin{align}
\boxed{f(x)=\int_1^{\infty}\frac{1}{x}\,dt=1}
\end{align}
```

For colored boxes use package **empheq**. For an example see section 53 on page 73.

45 Greek letters

\pmb

The **amsmath** package simulates a bold font for the greek letters, it writes a greek character twice with a small kerning. This is done with the macro `\pmb{<letter>}`. The `\mathbf{<character>` doesn't work with lower greek character.

α	$\boldsymbol{\alpha}$
β	$\boldsymbol{\beta}$
γ	$\boldsymbol{\gamma}$
δ	$\boldsymbol{\delta}$
ϵ	$\boldsymbol{\epsilon}$
...	...

46 Miscellenous commands

There are several commands which can be used in mathmode:

\overset
\underset
\boxed

Some examples are shown in table 19.

`$\underset{under}{baseline}$`	$\underset{under}{baseline}$
`$\overset{over}{baseline}$`	$\overset{over}{baseline}$

Table 19: Different Mathcommands

`\underset` is a useful macro for having limits under non operators (see section 69).

Part III
Other Packages

47 amsopn.sty

With this package it is very easy to declare new math operators, which are written in upright mode:

$\underset{s=p}{Res}\quad\underset{s=p}{\operatorname{Res}}$

```
\documentclass[10pt]{article}
\usepackage{amsmath}
\usepackage{amsopn}
\DeclareMathOperator{\Res}{Res}
\begin{document}
$\underset{s=p}{Res}\quad\underset{s=p}{\Res}$
\end{document}
```

Table 20 shows the predefined operatornames of **amsopn**.

`\arccos`	$\arccos$	`\arcsin`	$\arcsin$	`\arctan`	$\arctan$
`\arg`	$\arg$	`\cos`	$\cos$	`\cosh`	$\cosh$
`\cot`	$\cot$	`\coth`	$\coth$	`\csc`	$\csc$
`\deg`	$\deg$	`\det`	$\det$	`\dim`	$\dim$
`\exp`	$\exp$	`\gcd`	$\gcd$	`\hom`	$\hom$
`\inf`	$\inf$	`\injlim`	$\injlim$	`\ker`	$\ker$
`\lg`	$\lg$	`\lim`	$\lim$	`\liminf`	$\liminf$
`\limsup`	$\limsup$	`\ln`	$\ln$	`\log`	$\log$
`\max`	$\max$	`\min`	$\min$	`\Pr`	$\Pr$
`\projlim`	$\projlim$	`\sec`	$\sec$	`\sin`	$\sin$
`\sinh`	$\sinh$	`\sup`	$\sup$	`\tan`	$\tan$
`\tabh`	$\tanh$				

Table 20: The predefined operators of `amsopn.sty`

48 bm.sty

By default the math macro **mathbf** writes everything bold and in upright mode $\mathbf{y=f(x)}$ (`$\mathbf{y=f(x)$`), but it should be in italic mode especially for variables $\boldsymbol{y=f(x)}$ (`$\bm{y=f(x)$`). For writing a whole formula in bold have a look at section 22 on page 41.

49 accents.sty

If you want to write for example an underlined M, then you can do it as

`\underline{$M$}`	$\underline{M}$
`\underbar{$M$}`	$\underline{M}$
`$\underaccent{\bar}{M}$`	$\underaccent{\bar}{M}$

As seen, there is no difference in `\underline` and `\underbar`. For some reasons it may be better to use the package `accents.sty` with the `\underaccents` macro.

50 esint.sty

This is a very useful package when you want nice double or triple integral or curve integral symbols. The ones from `wasysym`[23] are not the best. `esint`[24] supports the following symbols:

$$\backslash\text{int} : \int \tag{50.1}$$

$$\backslash\text{iint} : \iint \tag{50.2}$$

$$\backslash\text{iiintop} : \iiintop \tag{50.3}$$

$$\backslash\text{iiiintop} : \iiiintop \tag{50.4}$$

$$\backslash\text{dotsintop} : \dotsintop \tag{50.5}$$

$$\backslash\text{ointop} : \ointop \tag{50.6}$$

$$\backslash\text{oiint} : \oiint \tag{50.7}$$

$$\backslash\text{sqint} : \sqint \tag{50.8}$$

$$\backslash\text{sqiint} : \sqiint \tag{50.9}$$

$$\backslash\text{ointctrclockwise} : \ointctrclockwise \tag{50.10}$$

$$\backslash\text{ointclockwise} : \ointclockwise \tag{50.11}$$

$$\backslash\text{varointclockwise} : \varointclockwise \tag{50.12}$$

$$\backslash\text{varointctrclockwise} : \varointctrclockwise \tag{50.13}$$

[23] CTAN://macros/latex/contrib/wasysym/
[24] CTAN://macros/latex/contrib/esint/

$$\backslash\text{fint} : \fint \tag{50.14}$$

$$\backslash\text{varoiint} : \oiint \tag{50.15}$$

$$\backslash\text{landupint} : \fint \tag{50.16}$$

$$\backslash\text{landdownint} : \fint \tag{50.17}$$

51 bigdelim.sty

This is a very useful package together with the **multirow.sty** package. In the following example we need additional parentheses for a different number of rows. This is also possible with the **array** environment, but not as easy as with **bigdelim.sty**. The trick is that you need one separatee column for a big delimiter, but with empty cells in all rows, which the delimiter spans.

$$\begin{pmatrix}
 & x_{11} & x_{12} & \dots & x_{1p} & \left.\vphantom{\begin{matrix}x\\x\\x\\x\end{matrix}}\right\}\text{some text} \\
\text{text}\left[\vphantom{\begin{matrix}x\\x\\x\\x\\x\end{matrix}}\right. & x_{21} & x_{22} & \dots & x_{2p} & \\
 & \vdots & & & & \\
 & x_{n_1 1} & x_{n_1 2} & \dots & x_{n_1 p} & \\
 & x_{n_1+1,1} & x_{n_1+1,2} & \dots & x_{n_1+1,p} & \left.\vphantom{\begin{matrix}x\\x\\x\end{matrix}}\right\}\text{some more text} \\
 & \vdots & & & & \\
 & x_{n_1+n_2,1} & x_{n_1+n_2,2} & \dots & x_{n_1+n_2,p} & \\
 & \vdots & & & &
\end{pmatrix}$$

```latex
\[
  \begin{pmatrix}
    & x_{11} & x_{12} & \dots & x_{1p} & \rdelim\}{4}{3cm}[some text]\\
    \ldelim[{5}{1cm}[text] & x_{21} & x_{22} & \dots & x_{2p} \\
    & \vdots\\
    & x_{n_1 1}& x_{n_1 2} & \dots & x_{n_1 p}\\
    & x_{n_1+1,1}&x_{n_1+1,2} & \dots & x_{n_1+1, p} &
       \rdelim\}{3}{3cm}[some more text]\\
    & \vdots\\
    & x_{n_1+n_2, 1} & x_{n_1+n_2,2} & \dots & x_{n_1+n_2,p}\\
    & \vdots \\
  \end{pmatrix}
\]
```

As seen in the above listing the left big delimiter is placed in the first column, all other rows start with second column. It is possible to use all columns above and below the delimiter. For the **array** environment there must be two more columns defined, in case of a big delimiter left and right. The syntax of **\ldelim** and **\rdelim** is:

```
\ldelim<delimiter>{<n rows>}{<added horizontal space>}[<text>]
\rdelim<delimiter>{<n rows>}{<added horizontal space>}[<text>]
```

Any delimiter which is possible for the `\left` or `\right` command are allowed, f.ex.: “`()[]{}|`”. The text is an optional argument and always typeset in text mode.

52 braket.sty

It is available at CTAN://macros/latex/contrib/other/misc/braket.sty and provides several styles for writing math expressions inside brakets. For example:

$$\left\{ x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}\right\}$$

```
\[ \left\{ x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}\right\} \]
```

looks not quit right and it is not really easy to get the first vertical line in the same size as the outer braces. Some solution maybe using `\vphantom`:

$$\left\{\vphantom{\frac{5}{3}}x\in\mathbf{R} \right|\left. 0<{|x|}<\frac{5}{3}\right\}$$

```
\[
\left\{\vphantom{\frac{5}{3}}x\in\mathbf{R} \right|\left. 0<{|x|}<\
    frac{5}{3}\right\}
\]
```

`braket.sty` has the macros

```
\Bra{<math expression>}
\Ket{<math expression>}
\Braket{<math expression>}
\Set{<math expression>}
```

and the same with a leading lower letter, which are not really interesting.

$$\left\langle x\in\mathbf{R}|0<|x|<\frac{5}{3}\right|$$

$$\left| x\in\mathbf{R}|0<|x|<\frac{5}{3}\right\rangle$$

$$\left\langle x\in\mathbf{R}\,\middle|\,0<\middle|x\middle|<\frac{5}{3}\right\rangle$$

$$\left\{\, x\in\mathbf{R}\,\middle|\,0<|x|<\frac{5}{3}\,\right\}$$

```
\[ \Bra{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
\[ \Ket{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
\[ \Braket{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
\[ \Set{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
```

The difference between the `\Set` and the `\Braket` macro is the handling of the vertical lines. In `\Set` only the first one gets the same size as the braces and in `\Braket` all.

$$\left\langle \phi \middle| \frac{\partial^2}{\partial t^2} \middle| \psi \right\rangle$$

```
\[  \Braket{ \phi | \frac{\partial^2}{\partial t^2} | \psi }\]
```

`\Bra` and `\Ket` do nothing with the inner vertical lines.

53 empheq.sty

This package supports different frames for math environments of the `amsmath` package. It doesn't support the environments `equation` and `eqnarray` from standard LaTeX.

With the optional argument of the environment `empheq` the preferred box type can be specified. A simple one is `\fbox`

$$\boxed{f(x) = \int_1^{\infty} \frac{1}{x}\,dt = 1} \tag{53.1}$$

```
\begin{empheq}[boxtype=\fbox]
\begin{align}
 f(x)=\int_1^{\infty}\frac{1}{x}\,dt=1
\end{align}
\end{empheq}
```

The same is possible with the macro `\colorbox`:

$$f(x) = \int_1^{\infty} \frac{1}{x}\,dt = 1 \tag{53.2}$$

```
\begin{empheq}[boxtype={\fboxsep=10pt\colorbox{yellow}}]
\begin{align}
 f(x)=\int_1^{\infty}\frac{1}{x}\,dt=1
\end{align}
\end{empheq}
```

The key **boxtype** can hold any possible LaTeX command sequence. Boxing subequations is also no problem, the **empheq** environment works in the same way:

$$\boxed{\begin{aligned} f(x) &= \int_1^{\infty}\frac{1}{x}\,dt = 1 \qquad (53.3a)\\ f(x) &= \int_2^{\infty}\frac{1}{x}\,dt = 0.25 \qquad (53.3b)\end{aligned}}$$

```
\begin{empheq}[boxtype={\fboxsep=10pt\colorbox{cyan}}]
\begin{subequations}
\begin{align}
 f(x) & =\int_1^{\infty}\frac{1}{x}\,dt=1\\
 f(x) & =\int_2^{\infty}\frac{1}{x}\,dt=0.25
\end{align}
\end{subequations}
\end{empheq}
```

For more information on **empheq** have a look at the documentation of the package which is available at any CTAN server.

54 exscale.sty

The following formula is written with the default fontsize where everything looks more or less well:

$$\int_{-1}^{+1}\frac{f(x)}{\sqrt{1-x^2}}\,dx \approx \frac{\pi}{n}\sum_{i=1}^{n} f\left(\cos\left(\frac{2i-1}{2n}\right)\right)$$

Writing the same with the fontsize **\huge** gives a surprising result, which belongs to the historical development of LaTeX, the int- and sum-symbols are not stretched. This extreme fontsize is often needed for slides and not only written „just for fun“.

$${}^{+1}_{-1}\,\frac{f(x)}{\sqrt{1-x^2}}\,dx \approx \frac{\pi}{n}\;{}^{n}_{i=1}\, f \quad \cos \quad \frac{2i-1}{2n}$$

Using the **exscale.sty**[25] package, which should be part of any local TeX installation, all symbols get the right size.

[25] CTAN://macros/latex/base/

$$\int_{-1}^{+1} \frac{f(x)}{\sqrt{1-x^2}}\,dx \approx \frac{\pi}{n}\sum_{i=1}^{n} f\left(\cos\left(\frac{2i-1}{2n}\right)\right)$$

55 `eucal.sty` and `euscript.sty`

This packages should be part of your local TEX installation, because they come with the amsmath packages. Otherwise get them from CTAN[26]. They support a scriptwriting of only upper letters

`\mathscr{...}` $\mathscr{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$

Read the documentation of the docs for the interdependence to the `\mathcal` command. For the above example the package `eucal.sty` was loaded with the option `mathscr`.

56 amscd - Commutative Diagrams

`amscd.sty` is part of the amsmath-bundle or available at CTAN[27] and has no options for the `\usepackage` command. `amscd.sty` does not support diagonal arrows but is much more easier to handle than the complex pstricks- or the xypic-package. On the other hand simple diagrams can be written with the array environment or look at [13].

$$\begin{CD}
R\times S\times T @>\text{restriction}>> S\times T \\
@VprojVV @VVprojV \\
R\times S @<<\text{inclusion}< S
\end{CD}$$

```
\[
\begin{CD}
  R\times S\times T @>\text{restriction}>> S\times T \\
        @VprojVV                          @VVprojV \\
  R\times S       @<<\text{inclusion}<     S
\end{CD}
\]
```

[26] CTAN://fonts/amsfonts/latex/euscript.sty
[27] CTAN://macros/latex/required/amslatex/math/amscd.dtx

57 xypic

The `xymatrix` macro is part of the `xypic`-package[28] which can be loaded with several options which are not so important.[29].

$$\begin{tikzcd} A \arrow[d, no head, squiggly] \arrow[dr, no head] & B & C \\ D \arrow[r, no head, dotted] & E \arrow[r, no head, squiggly] & F \arrow[dl, no head, squiggly] \\ G & H & I \end{tikzcd} \tag{57.1}$$

This matrix was created with

```
\[
\xymatrix{ A\POS [];[d]**\dir {~},[];[dr]**\dir {-} & B & C\\
 D & E\POS [];[l]**\dir {.},[];[r]**\dir {~} & F\POS [];[dl]**\dir {~}\\
 G & H & I}
\]
```

[28] CTAN://macros/generic/diagrams/xypic/xy-3.7/

[29] For more information look at the style file `xy.sty`, which is often saved in `/usr/share/texmf/tex/generic`.

Part IV
Special Symbols

In this section there are only those symbols defined, which are not part of the list of all available symbols: CTAN://info/symbols/comprehensive/symbols-a4.pdf.

58 Integral symbols

Name	Symbol
`\dashint`	$\dashint$
`\ddashint`	$\ddashint$
`\clockint`	$\clockint$
`\counterint`	$\counterint$

For all new integral symbols limits can be used in the usual way:

$$\ddashint_0 1=\dashint_1 0<\oint\limits_{-\infty}^\infty = \clockint\counterint_A \tag{58.1}$$

```
\ddashint_01=\dashint_10<\oint\limits_{-\infty}^\infty = \clockint\
    counterint_A
```

Put the following definitions into the preamble to use one or all of these new integral symbols.

```
\def\Xint#1{\mathchoice
   {\XXint\displaystyle\textstyle{#1}}%
   {\XXint\textstyle\scriptstyle{#1}}%
   {\XXint\scriptstyle\scriptscriptstyle{#1}}%
   {\XXint\scriptscriptstyle\scriptscriptstyle{#1}}%
   \!\int}
\def\XXint#1#2#3{{\setbox0=\hbox{$#1{#2#3}{\int}$}
     \vcenter{\hbox{$#2#3$}}\kern-.5\wd0}}
\def\ddashint{\Xint=}
\def\dashint{\Xint-}
\def\clockint{\Xint\circlearrowright} % GOOD!
\def\counterint{\Xint\rotcirclearrowleft} % Good for Computer Modern
    !
\def\rotcirclearrowleft{\mathpalette{\RotLSymbol{-30}}\
    circlearrowleft}
\def\RotLSymbol#1#2#3{\rotatebox[origin=c]{#1}{$#2#3$}}
```

59 Harpoons

LaTeX knows no stretchable harpoon symbols, like `\xrightarrow`. The following code defines several harppon symbols.

`\xrightharpoondown`
`\xrightharpoonup`
`\xleftharpoondown`
`\xleftharpoonup`
`\xleftrightharpoons`
`\xrightleftharpoons`

```
\def\rightharpoondownfill@{%
 \arrowfill@\relbar\relbar\rightharpoondown}
\def\rightharpoonupfill@{%
 \arrowfill@\relbar\relbar\rightharpoonup}
\def\leftharpoondownfill@{%
 \arrowfill@\leftharpoondown\relbar\relbar}
\def\leftharpoonupfill@{%
 \arrowfill@\leftharpoonup\relbar\relbar}
\newcommand{\xrightharpoondown}[2][]{%
 \ext@arrow 0359\rightharpoondownfill@{#1}{#2}}
\newcommand{\xrightharpoonup}[2][]{%
 \ext@arrow 0359\rightharpoonupfill@{#1}{#2}}
\newcommand{\xleftharpoondown}[2][]{%
 \ext@arrow 3095\leftharpoondownfill@{#1}{#2}}
\newcommand{\xleftharpoonup}[2][]{%
 \ext@arrow 3095\leftharpoonupfill@{#1}{#2}}
\newcommand{\xleftrightharpoons}[2][]{\mathrel{%
 \raise.22ex\hbox{%
  $\ext@arrow 3095\leftharpoonupfill@{\phantom{#1}}{#2}$}%
 \setbox0=\hbox{%
  $\ext@arrow 0359\rightharpoondownfill@{#1}{\phantom{#2}}$}%
 \kern-\wd0 \lower.22ex\box0}%
}
\newcommand{\xrightleftharpoons}[2][]{\mathrel{%
 \raise.22ex\hbox{%
  $\ext@arrow 3095\rightharpoonupfill@{\phantom{#1}}{#2}$}%
 \setbox0=\hbox{%
  $\ext@arrow 0359\leftharpoondownfill@{#1}{\phantom{#2}}$}%
 \kern-\wd0 \lower.22ex\box0}%
}
```

`\xrightharpoondown[under]{over}`	$\underset{under}{\overset{over}{\rightharpoondown}}$
`\xrightharpoonup[under]{over}`	$\underset{under}{\overset{over}{\rightharpoonup}}$
`\xleftharpoondown[under]{over}`	$\underset{under}{\overset{over}{\leftharpoondown}}$
`\xleftharpoonup[under]{over}`	$\underset{under}{\overset{over}{\leftharpoonup}}$
`\xleftrightharpoons[under]{over}`	$\underset{under}{\overset{over}{\leftrightharpoons}}$
`\xrightleftharpoons[under]{over}`	$\underset{under}{\overset{over}{\rightleftharpoons}}$

Part V
Examples

60 Identity Matrix

There are several possibilities to write this matrix. Here is a solution with the default array environment.

$$\left(\begin{array}{ccccc}
1\\
 & 1 & & \text{{\huge 0}}\\
 & & 1\\
 & \text{{\huge 0}} & & 1\\
 & & & & 1\end{array}\right)$$

```
\[
\left(\begin{array}{ccccc}
1\\
 & 1 & & \text{{\huge{0}}}\\
 & & 1\\
 & \text{{\huge{0}}} & & 1\\
 & & & & 1\end{array}\right)
\]
```

61 Cases Structure

Sometimes it's better to use the array environment instead of amsmaths cases environment. To get optimal horizontal spacing for the conditions, there are two matrixes in series, one $3x1$ followed by $3x3$ matrix. To minimize the horizontal space around the variable z a

```
\addtolength{\arraycolsep}{-3pt}
```

is a useful command.

$$I(z)=\delta_{0}\left\{
\begin{array}{lcrcl}
D+z & \quad & -D & \le z\le & -p\\
D-\frac{1}{2}\left(p-\frac{z^{2}}{p}\right)
 & \quad & -p & \le z\le & p\\
D-z & \quad & p & \le z\le & D
\end{array}\right. \tag{61.1}$$

```
\addtolength{\arraycolsep}{-3pt}
I(z)=\delta_{0}\left\{%
 \begin{array}{lcrcl}
  D+z & \quad & -D & \le z\le & -p\\
  D-\frac{1}{2}\left(p-\frac{z^{2}}{p}\right)%
    & \quad & -p & \le z\le & \phantom{-}p\\
  D-z & \quad & p & \le z\le & \phantom{-}D
 \end{array}\right.
\end{equation}
```

The `\phantom` command replaces exactly that place with whitespace which the argument needs .

62 Arrays

There is a general rule that a lot of mathematical stuff should be divided in smaller pieces. But sometimes it's difficult to get a nice horizontal alignment when splitting a formula. The following ones uses the `array` environment to get a proper alignment.

62.1 Quadratic Equation

$$
\begin{array}{rcll}
y & = & x^{2}+bx+c\\
 & = & x^{2}+2\cdot{\displaystyle\frac{b}{2}x+c}\\
 & = & \underbrace{x^{2}+2\cdot\frac{b}{2}x+\left(\frac{b}{2}\right)^{2}}_{\qquad\left(x+{\displaystyle \frac{b}{2}}\right)^{2}}-{\displaystyle\left(\frac{b}{2}\right)^{2}+c}\\
 & = & \left(x+{\displaystyle \frac{b}{2}}\right)^{2}-\left({\displaystyle\frac{b}{2}}\right)^{2}+c & \left|+\left({\displaystyle\frac{b}{2}}\right)^{2}-c\right.\\
y+\left({\displaystyle \frac{b}{2}}\right)^{2}-c & = & \left(x+{\displaystyle\frac{b}{2}}\right)^{2} & \left|(\textrm{Scheitelpunktform})\right.\\
y-y_{S} & = & (x-x_{S})^{2}\\
S(x_{S};y_{S}) & \,\textrm{bzw.}\, & S\left(-{\displaystyle\frac{b}{2};\,\left({\displaystyle \frac{b}{2}}\right)^{2}-c}\right)
\end{array}
\tag{62.1}
$$

```
\begin{equation}
\begin{array}{rcll}
y & = & x^{2}+bx+c\\
  & = & x^{2}+2\cdot{\displaystyle\frac{b}{2}x+c}\\
  & = & \underbrace{x^{2}+2\cdot\frac{b}{2}x+\left(\frac{b}{2}\right)^{2}}-{\displaystyle%
 \left(\frac{b}{2}\right)^{2}+c}\\
 &  & \qquad\left(x+{\displaystyle \frac{b}{2}}\right)^{2}\\
 & = & \left(x+{\displaystyle \frac{b}{2}}\right)^{2}-\left({\displaystyle%
\frac{b}{2}}\right)^{2}+c & \left|+\left({\displaystyle%
\frac{b}{2}}\right)^{2}-c\right.\\
y+\left({\displaystyle \frac{b}{2}}\right)^{2}-c & = & \left(x+{\displaystyle%
\frac{b}{2}}\right)^{2} & \left|(\textrm{Scheitelpunktform})\right.\\
y-y_{S} & = & (x-x_{S})^{2}\\
S(x_{S};y_{S}) & \,\textrm{bzw.}\, & S\left(-{\displaystyle%
\frac{b}{2};\,\left({\displaystyle \frac{b}{2}}\right)^{2}-c}\right)
\end{array}
\end{equation}
```

62.2 Vectors and Matrices

$$
\begin{array}{rcl}
\underline{RS} & = & \left(\begin{array}{cccccccc}
01 & a4 & 55 & 87 & 5a & 58 & db & 9e\\
a4 & 56 & 82 & f3 & 1e & c6 & 68 & e5\\
02 & a1 & fc & c1 & 47 & ae & 3d & 19\\
a4 & 55 & 87 & 5a & 58 & db & 9e & 03\end{array}\right)\\
\\
\left(\begin{array}{c}
s_{i,0}\\
s_{i,1}\\
s_{i,2}\\
s_{i,3}
\end{array}\right) & = & \underline{RS}\cdot
\left(\begin{array}{c}
m_{8i+0}\\
m_{8i+1}\\
\cdots\\
m_{8i+6}\\
m_{8i+7}
\end{array}\right)\\
\\
S_{i} & = & \sum_{j=0}^{3}s_{i,j}\cdot2^{8j}\qquad i=0,1,...,k-1\\
\\
S & = & \left(S_{k-1},S_{k-2},...,S_{1},S_{0}\right)
\end{array}
\tag{62.2}
$$

```
\begin{equation}
\begin{array}{rcl}
\underline{RS} & = & \left(\begin{array}{cccccccc}
01 & a4 & 55 & 87 & 5a & 58 & db & 9e\\
a4 & 56 & 82 & f3 & 1e & c6 & 68 & e5\\
02 & a1 & fc & c1 & 47 & ae & 3d & 19\\
a4 & 55 & 87 & 5a & 58 & db & 9e & 03\end{array}\right)\\
\\
\left(\begin{array}{c}
 s_{i,0}\\
 s_{i,1}\\
 s_{i,2}\\
 s_{i,3}
\end{array}\right) & = & \underline{RS}\cdot%
\left(\begin{array}{c}
 m_{8i+0}\\
 m_{8i+1}\\
 \cdots\\
 m_{8i+6}\\
 m_{8i+7}
\end{array}\right)\\
\\
S_{i} & = & \sum_{j=0}^{3}s_{i,j}\cdot2^{8j}\qquad i=0,1,...,k-1\\
\\
S & = & \left(S_{k-1},S_{k-2},...,S_{1},S_{0}\right)
\end{array}
\end{equation}
```

62.3 Cases with (eqn)array environment

This solution is important when amsmath.sty couldn't be used.

$$\lim_{n->\infty}q^{n}=\left\{
\begin{array}{lr}
\textrm{divergent}\ & q\le-1\\
0 & |q|<1\\
1 & q=1\\
\infty & q>1
\end{array}\right.$$

```
$\lim_{n->\infty}q^{n}=\left\{%
 \begin{array}{lr}
  \textrm{divergent}\ & q\le-1\\
  0 & |q|<1\\
  1 & q=1\\
  \infty & q>1
 \end{array}\right.$
```

62.4 Arrays inside Arrays

The array environment is a powerful one because it can be nested in several ways:

$$\left(
\begin{array}{c@{}c@{}c}
\begin{array}{|cc|}\hline
a_{11} & a_{12} \\
a_{21} & a_{22} \\\hline
\end{array} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} &
\begin{array}{|ccc|}\hline
b_{11} & b_{12} & b_{13}\\
b_{21} & b_{22} & b_{23}\\
b_{31} & b_{32} & b_{33}\\\hline
\end{array} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} &
\begin{array}{|cc|}\hline
c_{11} & c_{12} \\
c_{21} & c_{22} \\\hline
\end{array} \\
\end{array}
\right)$$

```
\[
\left(
\begin{array}{c@{}c@{}c}
 \begin{array}{|cc|}\hline
  a_{11} & a_{12} \\
  a_{21} & a_{22} \\\hline
 \end{array} & \mathbf{0} & \mathbf{0} \\
 \mathbf{0} &
 \begin{array}{|ccc|}\hline
  b_{11} & b_{12} & b_{13}\\
  b_{21} & b_{22} & b_{23}\\
  b_{31} & b_{32} & b_{33}\\\hline
 \end{array} & \mathbf{0} \\
 \mathbf{0} & \mathbf{0} &
 \begin{array}{|cc|}\hline
  c_{11} & c_{12} \\
  c_{21} & c_{22} \\\hline
 \end{array} \\
\end{array}
\right)
\]
```

$$
Y^1 = \begin{array}{c}
\left[\begin{array}{rrrr}
0 & 0 & 1 & 0\\
1 & 0 & 1 & 0\\
1 & 1 & 1 & 1
\end{array}\right]\\[3ex]\hline
\begin{array}{rrrr}
2 & 1 & 3 & 1
\end{array}
\end{array}
$$

```latex
\[
Y^1=
\begin{array}{c}
 \null\\[1ex]% only vor vertical alignment
 \left[\begin{array}{rrrr}
  0 & 0 & 1 & 0\\
  1 & 0 & 1 & 0\\
  1 & 1 & 1 & 1
 \end{array}\right]\\[3ex]\hline
 \begin{array}{rrrr}
% \hdotsfor{4}\\%( needs amsmath) instead of \\[3ex]\hline
  2 & 1 &3 & 1
 \end{array}
\end{array}
\]
```

63 Over- and underbraces

Overlapping under- and overbraces like $\underbrace{\qquad\quad}_{u1}\!\overbrace{\qquad\qquad}^{o}\!\underbrace{\qquad\quad}_{u2}$ needs some tricky code, because we cannot have parts of the argument inside **overbrace** and also **underbrace** . The following equation 63.1 is an example for such a construction:

$$
\begin{aligned}
y &= 2x^2 - 3x + 5 && (63.1)\\
&= 2\left(x^2 - \frac{3}{2}x + \overbrace{\left(\frac{3}{4}\right)^2 - \left(\frac{3}{4}\right)^2}^{=0} + \frac{5}{2}\right) && (63.2)\\
&= 2\left(\left(x - \frac{3}{4}\right)^2 + \frac{31}{16}\right) && (63.3)\\
y - \frac{31}{8} &= 2\left(x - \frac{3}{4}\right)^2 && (63.4)
\end{aligned}
$$

```
\begin{align}\label{eq:pqFormel}
y &= 2x^2 -3x +5\\\nonumber
  & \hphantom{= \ 2\left(x^2-\frac{3}{2}\,x\right. }%
        \textcolor{blue}{%
             \overbrace{\hphantom{+\left(\frac{3}{4}\right)^2- %
                 \left(\frac{3}{4}\right)^2}}^{=0}}\\[-11pt]
  &= 2\left(\textcolor{red}{%
     \underbrace{%
         x^2-\frac{3}{2}\,x + \left(\frac{3}{4}\right)^2}%
     }%
     \underbrace{%
        - \left(\frac{3}{4}\right)^2 + \frac{5}{2}}%
     \right)\\
  &= 2\left(\qquad\textcolor{red}{\left(x-\frac{3}{4}\right)^2}
     \qquad + \ \frac{31}{16}\qquad\right)\\
y\textcolor{blue}{-\frac{31}{8}}
  &= 2\left(x\textcolor{cyan}{-\frac{3}{4}}\right)^2%
\end{align}
```

64 Integrals

The *first theorem of Green* is:

$$\underset{\mathcal{G}}{\iiint}\left[u\nabla^2 v+(\nabla u,\nabla v)\right]d^3V=\underset{\mathcal{S}}{\oiint}u\frac{\partial v}{\partial n}d^2A$$

The *second theorem of Green* is:

$$\underset{\mathcal{G}}{\iiint}\left[u\nabla^2 v-v\nabla^2 u\right]d^3V=\underset{\mathcal{S}}{\oiint}\left(u\frac{\partial v}{\partial n}-v\frac{\partial u}{\partial n}\right)d^2A$$

They are both written with the **esint.sty** package[30], which gives nice integral symbols. The LaTeX-code for the first equation is:

```
\underset{%              for the integral
  {\cal G}\ \ \ %        the limit with space to move it left
}%                       end of the limit
{\iiint }%               the triple integral - end of \underset
\left[%                  bracket open
  u\nabla ^{2}v+\left(% parentheses open
\nabla u,\nabla v\right)% close
\right]%                 close bracket
d^{3}V=%                 end left part of the equation
\underset{%              see above
  {\cal S}\ \ \ %        s.a.
```

[30] See Section 64.

```
}%                          s.a
{\oiint }%                  the line integral
u\Q {v}{n}d^{2}A%           end right side
```

with the following definition in the preamble for the partial derivation:

```
\def\Q#1#2{\frac{\partial#1}{\partial #2}}
```

which makes things easier to write.

65 Vertical Alignment

Sometimes it maybe useful to have a vertical alignment over the whole page with a mix of formulas and text. Section 42 shows the use of `\intertext`. There is another trick to get all formulas vertical aligned. Let's have the following formulas distributed over the whole page:

$$\begin{aligned} f(x) &= a \\ g(x) &= x2 - 4x \\ f(x) - g(x) &= x2 + x3 + x \\ g &= x2 + x3 + x4 + x5 + b \end{aligned}$$

They all have a different length of the left and right side. Now we want to write some text and other objects between them, but let the alignment untouched. We choose the longest left and the longest right side and take them for scaling with the `\hphantom` command:

```
\hphantom{f(x)-g(x)} & \hphantom{= x2+x3+x4+x5+b}
```

This is the first (empty) line in every equation where now all other lines are aligned to this one. For example:

---

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

$$f(x) = a \tag{65.1}$$

$$g(x) = x2 - 4x \tag{65.2}$$

a	b	1
c	d	2

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

$$f(x) - g(x) = x2 + x3 + x \tag{65.3}$$

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

$$g(x) = x2 + x3 + x4 + x5 + b \tag{65.4}$$

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

---

The phantom line is empty but leaves the vertical space for a line. This could be corrected with decreasing the `\abovedisplayshortskip` length and restoring them after the whole sequence of commands. The code of the above looks like:

```
\newcommand{\x}{blah blah blah blah blah blah blah blah }
\addtolength{\abovedisplayshortskip}{-1cm} % decrease the skip
\addtolength{\abovedisplayskip}{-1cm}
\x\x\x\x\x
\begin{align}
\hphantom{f(x)-g(x)} & \hphantom{= x2+x3+x4+x5+b}\nonumber\\
f(x) &= a\\
g(x) &= x2-4x
\end{align}
\begin{center}
\begin{tabular}{cc|c}
 a & b & 1\\\hline
 c & d & 2
\end{tabular}
\end{center}
\x\x\x\x\x
\begin{align}
\hphantom{f(x)-g(x)} & \hphantom{= x2+x3+x4+x5+b}\nonumber\\
f(x)-g(x) &= x2+x3+x
\end{align}
\x\x\x\x\x

\begin{align}
\hphantom{f(x)-g(x)} & \hphantom{= x2+x3+x4+x5+b}\nonumber\\
g(x) &= x2+x3+x4+x5+b
\end{align}
\x\x\x\x\x
% restore old values
\addtolength{\abovedisplayshortskip}{1cm}
\addtolength{\abovedisplayskip}{1cm}
```

66 Node connections

This is a typical application for PSTricks and it needs the package `pst-node` and doesn't work with `pdflatex`. Use `VTeX`, `ps4pdf` or `ps2pdf`.

Die Bindungsenergie im Tröpfchenmodell setzt sich aus folgenden Teilen zusammen:

- dem Oberflächenanteil
- Dem Volumenanteil,

$$E = a_vA + -a_fA^{2/3} + -a_c\frac{Z(Z-1)}{A^{1/3}} + -a_s\frac{(A-2Z)^2}{A} + E_p \tag{66.1}$$

- dem Coulomb-Anteil
- der Symmetrieenergie
- sowie einem Paarbildungsbeitrag.

```
\psset{nodesep=3pt}
\newrgbcolor{lila}{0.6 0.2 0.5}
\newrgbcolor{darkyellow}{1 0.9 0}
Die Bindungsenergie im Tröpfchenmodell setzt sich aus
folgenden Teilen zusammen:
\begin{itemize}
\item dem \rnode{b}{Oberflächenanteil}
\item Dem \rnode{a}{Volumenanteil},\\[1cm]
\def\xstrut{\vphantom{\frac{(A)^1}{(B)^1}}}
\begin{equation}
E =
\rnode[t]{ae}{\psframebox*[fillcolor=darkyellow,
  linestyle=none]{\xstrut a_vA}} +
\rnode[t]{be}{\psframebox*[fillcolor=lightgray,
  linestyle=none]{\xstrut -a_fA^{2/3}}} +
\rnode[t]{ce}{\psframebox*[fillcolor=green,
  linestyle=none]{\xstrut -a_c\frac{Z(Z-1)}{A^{1/3}}}} +
\rnode[t]{de}{\psframebox*[fillcolor=cyan,
  linestyle=none]{\xstrut -a_s\frac{(A-2Z)^2}{A}}} +
\rnode[t]{ee}{\psframebox*[fillcolor=yellow,
  linestyle=none]{\xstrut E_p}}
\end{equation}\\[0.25cm]
\item dem \rnode{c}{Coulomb-Anteil}
```

```
\item der \rnode{d}{Symmetrieenergie}
\item sowie einem \rnode{e}{Paarbildungsbeitrag}.
\end{itemize}
\nccurve[angleA=-90,angleB=90]{->}{a}{ae}
\nccurve[angleB=45]{->}{b}{be}
\nccurve[angleB=-90]{->}{c}{ce}
\nccurve[angleB=-90]{->}{d}{de}
\nccurve[angleB=-90]{->}{e}{ee}
```

67 Formulas side by side

Sometimes it may be useful to have numbered formulas side by side like the following ones:

$\oint E ds = 0$	(67.1.a)	$\nabla \cdot B = 0$	(67.1.b)
$a = \frac{c}{d}$	(67.2.a)	$b = 1$	(67.2.b)
$c = 1$	(67.3.a)	$\int 2x dx = x2$	(67.3.b)

And again a default display formular:

$$F(x) = \int_0^\infty \frac{1}{x}\,dx \tag{67.4}$$

```
\begin{mtabular}{*{2}{m{0.35\linewidth}m{0.15\linewidth}}}
\begin{align*} \oint E ds=0 \end{align*} & \eqnCnt %
 & \begin{align*} \nabla\cdot B=0 \end{align*} & \eqnCnt[\label{blah}]\\
\begin{align*} a =\frac{c}{d} \end{align*} & \eqnCnt %
 & \begin{align*} b = 1 \end{align*} & \eqnCnt\\
\begin{align*} c =1 \end{align*} & \eqnCnt[\label{blub}]
 & \begin{align*} \int 2x dx = x2 \end{align*} & \eqnCnt
\end{mtabular}
```

The new environment `mtabular` has two arguments, one optional and one which is the same than the one from the `tabular` environment. With the option `long` it is possible to have all the formulas in a `longtable` environment, which allows a pagebreak. The new macro `\eqnCnt` controls the counting of these equations as subeuqations for one tabular line. This macro can have an optional argument for a label. At least it counts the equations. If the equation number is not centered to the forgoing equation, then it needs some more horizontal space in the tabular column.

`\eqnCnt[<optional label>]`

The vertical space is controlled by the length **mtabskip**, which is by default `-1.25cm` and can be modified in the usual way.

To define all these macros write into the preamble:

```
\usepackage{amsmath}
\newcounter{subequation}
%
\newlength\mtabskip\mtabskip=-1.25cm
%
\newcommand\eqnCnt[1][]{%
 \refstepcounter{subequation}%
 \begin{align}#1\end{align}%
 \addtocounter{equation}{-1}%
}
\def\mtabLong{long}
\makeatletter
\newenvironment{mtabular}[2][\empty]{%
 \def\@xarraycr{%
  \stepcounter{equation}%
  \setcounter{subequation}{0}%
  \@ifnextchar[\@argarraycr{\@argarraycr[\mtabskip]}%
 }
 \let\theoldequation\theequation%
 \renewcommand\theequation{\theoldequation.\alph{subequation}}
 \edef\mtabOption{#1}
 \setcounter{subequation}{0}%
 \tabcolsep=0pt
 \ifx\mtabOption\mtabLong\longtable{#2}\else\tabular{#2}\fi%
}{%
 \ifx\mtabOption\mtabLong\endlongtable\else\endtabular\fi%
 \let\theequation\theoldequation%
 \stepcounter{equation}
}
\makeatother
```

As seen in equation 67.3.a and eq.67.1.b, everything is nonsense ... And the following tabular is defined as lontable to enable pagebreaks.

$\oint Eds = 0$	(67.5.a)	$\nabla \cdot B = 0$	(67.5.b)
$a = \frac{c}{d}$	(67.6.a)	$b = 1$	(67.6.b)
$c = 1$	(67.7.a)	$\int 2xdx = x2$	(67.7.b)

$\oint Eds = 0$	(67.8.a)	$\nabla \cdot B = 0$	(67.8.b)
$a = \frac{c}{d}$	(67.9.a)	$b = 1$	(67.9.b)
$c = 1$	(67.10.a)	$\int 2xdx = x2$	(67.10.b)
$\oint Eds = 0$	(67.11.a)	$\nabla \cdot B = 0$	(67.11.b)
$a = \frac{c}{d}$	(67.12.a)	$b = 1$	(67.12.b)
$c = 1$	(67.13.a)	$\int 2xdx = x2$	(67.13.b)
$\oint Eds = 0$	(67.14.a)	$\nabla \cdot B = 0$	(67.14.b)
$a = \frac{c}{d}$	(67.15.a)	$b = 1$	(67.15.b)
$c = 1$	(67.16.a)	$\int 2xdx = x2$	(67.16.b)

As seen in equation 67.13.a and eq.67.11.b, everything is nonsense ...
And again a default display formular:

$$F(x) = \int_0^\infty \frac{1}{x}\,dx \tag{67.17}$$

```
\begin{mtabular}[long]{*{2}{m{0.375\linewidth}m{0.125\linewidth}}}
\begin{align*} \oint E ds=0 \end{align*} & \eqnCnt %
 & \begin{align*} \nabla\cdot B=0 \end{align*} & \eqnCnt\\
\begin{align*} a =\frac{c}{d} \end{align*} & \eqnCnt %
 & \begin{align*} b = 1 \end{align*} & \eqnCnt\\
\begin{align*} c =1 \end{align*} & \eqnCnt
 & \begin{align*} \int 2x dx = x2 \end{align*} & \eqnCnt\\

[ ... ]
```

List of Figures

List of Tables

References

[1] Paul W. Abrahams, Karl Berry, and Kathryn Hargreaves. *TEX for the Impatient*. http://tug.org/ftp/tex/impatient/book.pdf, 2003.

[2] Michael Downes. *Technical Notes on the amsmath package*. American Mathematical Society, ftp://ftp.ams.org/pub/tex/doc/amsmath/technote.pdf, 1999.

[3] Michael Downes. *Short Math Guide for LATEX*. American Mathematical Society, http://www.ams.org/tex/short-math-guide.html, 2002.

[4] Victor Eijkhout. *TEX by Topic*. http://www.eijkhout.net/tbt/, 1992.

[5] J. Anthony Fitzgerald. *Web Math Formulas Using TEX*. http://www.unb.ca/web/Sample/math/, 1997.

[6] Michel Goosens, Frank Mittelbach, and Alexander Samarin. *The LATEX Companion*. Addison Wesley, 13 edition, 1994.

[7] George Grätzer. *Math into LATEX*. Birkhäuser Boston, third edition, 2000.

[8] Donald E. Knuth. *The TEXbook*. Addison Wesley Professional, 21 edition, 1986.

[9] Donald E. Knuth, Tracy Larrabee, and Paul M. Roberts. *Mathematical Writing*. Stanford University, Computer Science Department, http://sunburn.stanford.edu/~knuth/papers/mathwriting.tex.gz, 1987.

[10] R. Kuhn, R. Scott, and L. Andreev. *An Introduction to using LATEX in the Harvard Mathematics Department*. Harvard University, Department of Mathematics, http://abel.math.harvard.edu/computing/latex/manual/texman.html.

[11] Sebastian Rahtz and Leonor Barroca. A style option for rotated objects in LATEX. *TUGboat Journal*, 13(2):156–180, July 1992.

[12] Steve Seiden. *Math cheat sheet*. TUG, http://www.tug.org/texshowcase/#math, 2000.

[13] Paul Taylor. *Commutative Diagrams in TEX*. Department of Computer Science, Queen Mary and Westfield College, http://www.dcs.qmw.ac.uk/~pt/diagrams/, 2000.

Index