

Mathebefehle für \LaTeX

Uwe Stöhr
uwestoehr@web.de

29. Oktober 2006, Version 3.09*

Diese Dokumentation unterliegt der
[GNU Free Documentation License](#).

Inhaltsverzeichnis

1	Einleitung	1
2	Allgemeine Hinweise	1
3	Grundlegende Funktionen	4
3.1	Exponenten und Indizes	4
3.2	Brüche	4
3.3	Wurzeln	5
3.4	Binomialkoeffizienten	6
3.5	Fallunterscheidungen	6
3.6	Verneinungen	6
3.7	Platzhalter	7
3.8	Linien	7
3.9	Fortsetzungspunkte	8
4	Matrizen	9

*Diese Dokumentation bezieht sich auf \LaTeX Version 1.4.3.

5	Klammern und Begrenzungszeichen	11
5.1	Vertikale Klammern und Begrenzungszeichen	11
5.1.1	Manuelle Klammergrößen	11
5.1.2	Automatische Klammergrößen	12
5.2	Horizontale Klammern	13
6	Pfeile	14
6.1	Horizontale Pfeile	14
6.2	Vertikale und diagonale Pfeile	15
7	Akzente	15
7.1	Akzente für ein Zeichen	15
7.2	Akzente für Operatoren	16
7.3	Akzente für mehrere Zeichen	17
8	Leerraum	17
8.1	Vordefinierter Leerraum	17
8.2	Variabler Leerraum	18
8.3	Leerraum neben eingebetteten Formeln	19
9	Boxen und Rahmen	20
9.1	Boxen mit Rahmen	20
9.2	Boxen ohne Rahmen	22
9.3	Farbige Boxen	22
9.4	Absatzboxen	24
10	Operatoren	26
10.1	Große Operatoren	26
10.2	Grenzen	27
10.3	Binäre Operatoren	29
10.4	Selbst definierte Operatoren	29
11	Schriften	30
11.1	Schriftstile	30
11.2	Fett gedruckte Formeln	31
11.3	Schriftgrößen	31
12	Griechische Buchstaben	33
12.1	Kleine Buchstaben	33
12.2	Große Buchstaben	33
12.3	Fett gedruckte Buchstaben	33
13	Symbole	34
13.1	Mathematische Symbole	34
13.2	Sonstige Symbole	34

13.3	Das EURO-Symbol	34
14	Relationen	35
15	Funktionen	36
15.1	Vordefinierte Funktionen	36
15.2	Selbst definierte Funktionen	36
15.3	Grenzwerte	37
15.4	Modulo-Funktionen	38
16	Sonderzeichen	38
16.1	Sonderzeichen im mathematischen Text	38
16.2	Akzente im Text	38
16.3	Minuskelziffern	39
16.4	Sonstige Sonderzeichen	40
17	Formelstile	40
18	Mehrzeilige Formeln	41
18.1	Allgemeines	41
18.1.1	Eqnarray-Umgebung	41
18.1.2	Zeilenabstand	42
18.1.3	Spaltenabstand	43
18.1.4	Lange Formeln	43
18.1.5	Mehrzeilige Klammern	45
18.2	Align-Umgebungen	45
18.2.1	Standard align-Umgebung	46
18.2.2	Alignat-Umgebung	46
18.2.3	Flalign-Umgebung	46
18.3	Gather-Umgebung	47
18.4	Multline-Umgebung	47
18.5	Mehrzeilige Formelteile	48
18.6	Text in mehrzeiligen Formeln	49
19	Formelnummerierung	49
19.1	Allgemeines	49
19.2	Querverweise	50
19.3	Formeluntergliederung	50
19.4	Benutzerdefinierte Nummerierung	51
19.5	Nummerierung mit römischen Zahlen und Buchstaben	53
20	Benutzerdefinierte Befehle	53
20.1	Der Befehl \newcommand	54
20.2	Mathe-Makros	55

21 Diagramme	57
22 Tipps	59
22.1 Chemische Symbole und Reaktionsgleichungen	59
22.2 Negative Zahlen	59
22.3 Komma als Dezimaltrennzeichen	59
22.4 Physikalische Vektoren	60
22.5 Selbst definierte Brüche	60
22.6 Durchgestrichene Formeln	61
22.7 Formeln in Überschriften	62
22.7.1 Überschrift ohne Formel im Inhaltsverzeichnis	63
22.7.2 Überschrift mit Formel im Inhaltsverzeichnis $\sqrt{-1} = i$	63
22.8 Formeln im mehrspaltigen Text	63
22.9 Formeln mit Beschreibung der Variablen	64
22.10 Aufrechte kleine griechische Buchstaben	65
22.11 Textzeichen in Formeln	65
22.12 Symbole aus dem textcomp-Paket	66
22.13 Textformatierung	66
22.13.1 Das Paket soul	67
22.13.2 Das Paket ulem	68
22.14 Anführungszeichen	68
22.15 Textboxen	69
22.15.1 Rotierter Text	69
22.15.2 Skalierter Text	70
22.15.3 Spezielle Anwendungen	71
22.16 Vertikaler Leerraum im Text	71
22.17 Worttrennungen, Zeilen- und Seitenumbrüche	72
22.17.1 Worttrennungen	72
22.17.2 Zeilenumbrüche	73
22.17.3 Seitenumbrüche	74
22.18 Links in DVI- und PDF-Dokumenten	74
22.18.1 Verlinkte Querverweise	74
22.18.2 Links zu externen Objekten	75
A Typographische Hinweise	78
B Synonyme	79
Literatur	79
Stichwort- und Befehlsverzeichnis	80

1 Einleitung

Dieses Dokument ist eine Zusammenstellung und Erläuterung der wichtigsten \LaTeX -Befehle für mathematische Zeichen und Konstruktionen. Die Erläuterungen sind speziell auf die Benutzung der Befehle mit \LyX zugeschnitten. Das setzt voraus, dass bereits das Kapitel **Mathematische Formeln** des Benutzerhandbuchs gelesen worden ist.

Die meisten Zeichen und viele Konstruktionen sind auch über das Fenster **Mathe-Kontrollfläche**¹, das Menü **Einfügen**▷**Mathe** oder die **Mathe-Symboleiste**² aufrufbar. Wer viele Formeln zu schreiben hat, wird jedoch feststellen, dass man mit den \LaTeX -Befehlen viel schneller eine Formel setzen kann.

Sofern nicht anders angegeben, sind alle Befehle nur in der Formel-Umgebung von \LyX verfügbar. Um alle Befehle nutzen zu können, muss die Option **AMS-Mathe-Paket verwenden** im Menü **Dokument**▷**Einstellungen**▷**Mathe Optionen** aktiviert sein.

Es wird in diesem Dokument der Übersichtlichkeit halber auf die Auflistung aller $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX -Befehle³ verzichtet.

Die Menünamen und Tastenkürzel beziehen sich auf die Qt-Version von \LyX mit deutschem Menü. Die XForms- und GTK-Versionen von \LyX haben teilweise andere Menübezeichnungen.

2 Allgemeine Hinweise

Um eine in einer Textzeile eingebettete Formel zu erstellen, drückt man **Strg-m** oder den Formelknopf der Symbolleiste.

Um eine abgesetzte Formel zu erstellen, drückt man **Strg-M**.

Um eine abgesetzte Formel in eine eingebettete umzuwandeln, setzt man den Cursor in die Formel und drückt **Strg-M** oder benutzt das Menü **Bearbeiten**▷**Mathe**▷**Formelart ändern**. Auf die selbe Weise lässt sich eine eingebettete in eine abgesetzte Formel umwandeln.

Um Teile einer eingebetteten Formel in der Größe einer abgesetzten darzustellen, gibt man in die Formel den Befehl `\displaystyle` ein. Es erscheint ein neues blaues Kästchen, in das dann der Formelteil eingegeben wird.

¹In das Fenster gelangt man entweder durch einen Klick mit der rechten Maustaste in eine Formel oder über das Menü **Einfügen**▷**Mathe**▷**Mathe-Kontrollfläche**.

²Symboleisten können ein- und ausgeblendet werden, indem man mit der rechten Maustaste auf die Menüleiste oder eine Symbolleiste klickt.

³Eine Liste mit allen $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX -Befehlen findet sich in der Datei **amsguide.ps**, die Teil jeder \LaTeX -Standardinstallation ist.

In einer Tabelle dürfen nur eingebettete Formeln verwendet werden.

In den $\text{T}_{\text{E}}\text{X}$ -Modus gelangt man über den $\text{T}_{\text{E}}\text{X}$ -Knopf der Symbolleiste oder das Menü **Einfügen** \triangleright **TeX** (Tastenkürzel **Strg-I**).

In den $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Vorspann gelangt man über das Menü **Dokument** \triangleright **Einstellungen** \triangleright **LaTeX-Vorspann**.

Die meisten $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Befehle für mathematische Konstruktionen besitzen folgendes Schema:

\backslash Befehlsname[optionales Argument]{anzugebendes Argument}

Ein Befehl muss immer mit einem Backslash „ \backslash “ beginnen. Möchte man ein optionales Argument nicht angeben, müssen auch die zugehörigen eckigen Klammern weggelassen werden. Die geschweiften Klammern werden als $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Klammern bezeichnet. Hängt man in einer Formel an Befehlsnamen eine linke geschweifte Klammer an, erzeugt $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ automatisch eine $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Klammer. Ansonsten werden sie in Formeln mit dem Befehl $\backslash\{$ erzeugt. $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Klammern erscheinen in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ rot, im Gegensatz zu normalen geschweiften Klammern, die blau erscheinen. Im $\text{T}_{\text{E}}\text{X}$ -Modus sind für $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Klammern keine Befehle notwendig. $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Klammern erscheinen nicht im Ausdruck.

Werden Befehle ohne Argumente, wie z. B. Symbolbefehle, im $\text{T}_{\text{E}}\text{X}$ -Modus eingegeben, muss auf den Befehl *immer* ein Leerzeichen folgen. Dieses Leerzeichen erscheint nicht im Ausdruck, sondern dient $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ nur als Zeichen, dass der Befehl zu Ende ist. Soll auch im Ausdruck ein Leerzeichen erscheinen, muss auf den Befehl im normalen Text ein geschütztes Leerzeichen folgen.

Ein geschütztes Leerzeichen gibt man mit **Strg-Leertaste** ein.

Zur nachträglichen Bearbeitung von Matrizen, Fallunterscheidungen und mehrzeiligen Formeln kann man die Menüs **Bearbeiten** \triangleright **Mathe** und **Bearbeiten** \triangleright **Zeilen & Spalten** oder die **Tabellen-Symbolleiste** verwenden. Beim Vertauschen von Zeilen und Spalten mit Hilfe dieses Menüs wird immer die Spalte bzw. Zeile, in der der Cursor steht, mit der rechts daneben stehenden Spalte bzw. darunter liegenden Zeile vertauscht. Befindet sich der Cursor in der letzten Spalte bzw. Zeile, wird mit der links daneben stehenden Spalte bzw. darüber liegenden Zeile vertauscht.

Zum Schreiben von Text in einer Formel⁴ verwendet man den *mathematischen Textmodus*, in den man mit dem Tastenkürzel **Alt-m m** oder durch Eingabe des Befehls $\backslash\text{text}$ gelangt. Der Text erscheint in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ schwarz und lässt sich dadurch gut vom Rest der Formel unterscheiden. Im fertigen Dokument erscheint mathematischer Text im Gegensatz zu allen anderen Zeichen einer Formel nicht kursiv.

Hinweise für Text

Ein Zeilenumbruch kann mit **Strg-Enter** erzwungen werden.

⁴Bei mehrzeiligen Formeln wird der Befehl $\backslash\text{intertext}$ verwendet, siehe Kap. 18.6.

Um ein Wort in LyX zu markieren, kann man mit der linken Maustaste doppelt auf das Wort klicken.

Verwendet man die Qt-Version von LyX unter Windows, kann man Text aus anderen Windows-Anwendungen in LyX kopieren. Dazu markiert man den Text und lädt ihn mit **Strg-c** (Kopieren) **Strg-x** (Ausschneiden) in den Zwischenspeicher. In LyX setzt man den Cursor an die gewünschte Position und drückt die mittlere Maustaste bzw. das Mausrad. Beim Kopieren von Text aus LyX kann man wie bei normalen Windows-Programmen vorgehen: Den Text markieren und mit **Strg-c** in den Zwischenspeicher laden, anschließend mit **Strg-v** in ein anderes Programmfenster einfügen.

Zeichenerklärung

- Das Symbol⁵ $_$ steht für ein einzugebendes Leerzeichen.
- Ein Pfeil wie z.B. \rightarrow steht für den Druck der jeweiligen Pfeiltaste auf der Tastatur.

Verwendete Längeneinheiten

Einheit	Name / Beschreibung
mm	Millimeter
cm	Zentimeter
in	Inch / Zoll (1 in = 2,54 cm)
pt	Punkt (72,27 pt = 1 in)
pc	Pica (1 pc = 12 pt)
sp	scaled point (65536 sp = 1 pt)
bp	big point (72 bp = 1 in)
dd	Didot (72 dd \approx 37,6 mm)
cc	Cicero (1 cc = 12 dd)
ex	Höhe des Buchstabens x in der aktuellen Schrift
em	Breite des Buchstabens M in der aktuellen Schrift
mu	math unit (1 mu = $\frac{1}{18}$ em)

⁵Dieses sichtbare Leerzeichen erhält man über den Befehl `\textvisiblespace`, der im T_EX-Modus einzugeben ist.

3 Grundlegende Funktionen

3.1 Exponenten und Indizes

Indizes werden mit dem Unterstrich `_`, Exponenten mit dem Zirkumflexzeichen `^` erzeugt.

Befehl	Ergebnis
<code>B_V</code>	B_V
<code>B^V</code>	B^V
<code>B^_A</code>	B^A

Da der Zirkumflex ein Akzent ist, werden Selbstlaute⁶ akzentuiert statt hochgestellt. Um in diesem Fall Exponenten zu erzeugen, muss nach dem Zirkumflex ein Leerzeichen eingegeben werden, siehe letztes Beispiel.

3.2 Brüche

Brüche werden mit dem Befehl `\frac` erstellt. Die Zeichengröße wird automatisch angepasst, je nachdem, ob sich der Bruch in einer eingebetteten oder abgesetzten Formel befindet.

Mit dem Befehl `\dfrac` kann man einen Bruch erzeugen, der in der Größe einer abgesetzten Formel erscheint. Mit `\tfrac` erscheint der Bruch in der Größe einer eingebetteten Formel. Ein Beispiel:

Eine Zeile mit dem Bruch $\frac{1}{2}$, der mit dem Befehl `\frac` erstellt wurde.

Eine Zeile mit dem Bruch $\frac{1}{2}$, der mit dem Befehl `\dfrac` erstellt wurde.

Befehl	Ergebnis
<code>\frac_A\downarrow B</code>	$\frac{A}{B}$
<code>\dfrac_A\downarrow B</code>	$\frac{A}{B}$
<code>\dfrac_e^_ \frac_1\downarrow 2\downarrow 3</code>	$\frac{e^{\frac{1}{2}}}{3}$

⁶Je nach Tastaturdefinition werden auch andere Buchstaben akzentuiert.

Für Mehrfachbrüche gibt es den Befehl `\cfrac`. Dazu ein Beispiel:

mit `\frac` erstellt

$$\frac{A}{B + \frac{C + \frac{E}{F}}{D}}$$

mit `\cfrac` erstellt

$$\frac{A}{B + \frac{C + \frac{E}{F}}{D}}$$

`\cfrac` bewirkt, dass der Bruch immer in der Größe einer abgesetzten Formel erscheint, auch wenn er selbst Teil eines Bruches ist. `\cfrac` besitzt folgendes Befehlschema:

`\cfrac[Zählerposition]{Zähler}{Nenner}`

Zählerposition kann *l* oder *r* sein und legt fest, ob der Zähler links- oder rechtsbündig am Bruchstrich ausgerichtet wird. Wird keine Position angegeben, erscheint der Zähler wie gewohnt mittig.

Der Befehl für obiges Beispiel lautet:

`\cfrac{A \rightarrow \{ B + \cfrac{C + \cfrac{E \rightarrow \{ F \rightarrow \rightarrow \{ D`

Oftmals ist es am günstigsten `\cfrac` und `\frac` zu kombinieren:

$$\frac{A}{B + \frac{C + \frac{E}{F}}{D}}$$

Wie man eigene Brüche definiert, um die Bruchstrichdicke zu ändern oder Brüche mit Schrägstrich zu erstellen, wie z. B. $\frac{5}{31}$, ist in Kap. 22.5 erklärt.

3.3 Wurzeln

Quadratwurzeln werden mit `\sqrt` gesetzt, alle anderen Wurzeln mit dem Befehl `\root`.

Befehl	Ergebnis
<code>\sqrt{A-B}</code>	$\sqrt{A - B}$
<code>\root{3}\downarrow{A-B}</code>	$\sqrt[3]{A - B}$

Eine Quadratwurzel kann man auch mit `\root` erstellen, indem man das Feld über der Wurzel frei lässt.

Bei manchen Indizes ist der Abstand zur Wurzel zu gering, wie bei dieser Formel: $\sqrt[n]{B}$

Das β berührt die Wurzel. Um dies zu verhindern, verwendet man die Befehle `\leftroot` und `\uproot`, die folgendes Schema besitzen:

`\leftroot{Abstand}` bzw. `\uproot{Abstand}`

Abstand ist die Anzahl der Big Points (Einheit bp; 72 bp = 1 inch), um die der Index nach links bzw. nach oben verschoben werden soll. Die Befehle werden in den Index geschrieben. So ergibt der Befehl `\root{-1}\leftroot{-1}\uproot{2}\beta\rightarrow B` eine korrekt gesetzte Formel: $\sqrt[2]{B}$

3.4 Binomialkoeffizienten

Befehl	Ergebnis
<code>\binom{A}{B}</code>	$\binom{A}{B}$

Analog zu Brüchen (`\frac`) gibt es außer `\binom` noch die Befehle `\dbinom` und `\tbinom`, die folgendes Befehlsschema besitzen:

`\dbinom{Zähler}{Nenner}` bzw. `\tbinom{Zähler}{Nenner}`

3.5 Fallunterscheidungen

Befehl	Ergebnis
<code>\cases{A\rightarrow B>0}</code>	$\begin{cases} A & B > 0 \end{cases}$
<code>\cases{Strg-Enter}</code>	$\begin{cases} A & \text{für } x > 0 \\ B & \text{für } x = 0 \end{cases}$

Nach der Eingabe von `\cases` können mit **Strg-Enter** neue Zeilen erstellt werden.⁷

Der Befehl `\cases` ist auch über das Menü **Einfügen** \triangleright **Mathe** \triangleright **Cases-Umgebung** verfügbar.

3.6 Verneinungen

Durch die Eingabe von `\not` kann jedes beliebige Zeichen durchgestrichen dargestellt werden. Die Zeichen werden quasi mit einem Schrägstrich akzentuiert.

Befehl	Ergebnis
<code>\not=</code>	\neq
<code>\not\leq</code>	\nless
<code>\not\parallel</code>	\nparallel

⁷Umlaute in Formeln sind in Kap. 7.1 beschrieben.

Das letzte Beispiel zeigt, dass nicht alle Verneinungen gut aussehen. Deswegen gibt es für Einige spezielle Befehle (siehe Kap. 13.1 und 14).

3.7 Platzhalter

Möchte man z. B. Isotope⁸ darstellen, tritt folgendes Problem auf:

durch Hoch- und Tiefstellen erzeugte Indizes: ${}^{19}_9\text{F}$
 korrekte Indizes: ${}^{19}_9\text{F}$

Der kürzere Index wird immer unter oder über das erste Zeichen des längeren Index gesetzt. Um das zu verhindern gibt es den Befehl `\phantom`, der ein oder mehrere Phantomzeichen erstellt. Es erscheint ein kleines blaues Kästchen, dass von zwei roten Pfeilen überlagert wird. Die Pfeile geben an, dass die komplette Breite und Höhe des Kästcheninhalts als Leerraum erstellt wird. Phantomzeichen sind dementsprechend Platzhalter für die im Kästchen angegebenen Zeichen.

Befehl	Ergebnis
<code>{}^{19}_\rightarrow 9_\text{F}</code>	${}^{19}_9\text{F}$
<code>{}^{235}_\rightarrow 9_\text{F}</code>	${}^{235}_9\text{F}$
<code>\Lambda^{}_\rightarrow x_\text{MMt}</code>	Λ^x_{MMt}

Das letzte Beispiel zeigt, wie sich ein falsches Argument auswirkt.

Des Weiteren gibt es noch die Befehle `\vphantom` und `\hphantom`. `\vphantom` erzeugt nur Platz für die maximale Höhe der im Argument angegebenen Zeichen, nicht jedoch für deren Breite. Bei `\hphantom` wird nur Platz für die Breite der Argumentzeichen erstellt. Daher haben die Kästchen der beiden Befehle nur einen roten Pfeil.

Zum Beispiel erzeugt `\vphantom_a\int` Platz der Höhe des Integralzeichens, da dieses das größte Zeichen im Argument ist. Eine Beispielanwendung ist in Kap. 18.1.5 zu finden.

3.8 Linien

Befehl	Ergebnis
<code>\overline_A+B</code>	$\overline{A+B}$
<code>\underline_A+B</code>	$\underline{A+B}$
<code>\overline_\underline_A+B</code>	$\overline{\underline{A+B}}$

⁸Weiteres zu chemischen Symbolen steht in Kap. 22.1.

Beim letzten Beispiel spielt es keine Rolle, ob man zuerst `\overline` oder `\underline` eingibt.

Will man Ergebnisse doppelt unterstreichen, verwendet man zweimal hintereinander `\underline`.

Man kann bis zu sechs Striche über oder unter eine Formel setzen.

3.9 Fortsetzungspunkte

Es gibt zwei unterschiedliche Arten horizontaler Punkte. Für Aufzählungen verwendet man unten sitzende Punkte (`\ldots`), während man für Operationen Punkte braucht, die auf der selben Höhe wie die Operatoren sind (`\cdots`). Verwendet man den Befehl `\dots`, entscheidet \LaTeX anhand des nachfolgenden Zeichens, welche Punktart verwendet wird.

Befehl	Ergebnis
A_1, \dots, A_n	A_1, \dots, A_n
$A_1 + \dots + A_n$	$A_1 + \dots + A_n$
A_1, \ldots, A_n	A_1, \ldots, A_n
$A_1 + \cdots + A_n$	$A_1 + \cdots + A_n$
<code>\vdots</code>	\vdots
<code>\ddots</code>	\ddots
3×3 Matrix mit obigen Befehlen	$\begin{pmatrix} A_{11} & \cdots & A_{1m} \\ \vdots & \ddots & \vdots \\ A_{n1} & \cdots & A_{nm} \end{pmatrix}$

Die im Menü **Einfügen** \triangleright **Sonderzeichen** verfügbaren Fortsetzungspunkte sind `\ldots`.

Speziell für Matrizen gibt es Fortsetzungspunkte, die sich über mehrere Spalten erstrecken. Man erhält sie mit dem Befehl `\hdotsfor`, der folgendes Schema besitzt:

`\hdotsfor[Abstand]{Anzahl der Spalten}`

Anzahl der Spalten gibt an, über wie viele Spalten sich die Punkte erstrecken sollen. Abstand ist ein Faktor, der den Punktabstand festlegt.

In folgender 3×4 Matrix wurde in das erste Feld der zweiten Zeile der Befehl `\hdotsfor[2]{4}` eingegeben, um Fortsetzungspunkte mit doppelt so großem Punktabstand wie beim Befehl `\dots` zu erhalten:

$$\begin{pmatrix} A & B & C & D \\ \dots & \dots & \dots & \dots \\ q & w & e & r \end{pmatrix}$$

Es ist zu beachten, dass die Matrixfelder über die sich die Punkte erstrecken sollen leer sein müssen, ansonsten kommt es zu L^AT_EX-Fehlern.

Des Weiteren kann man mit dem Befehl `\dotfill` den Rest einer Zeile mit Punkten füllen. Die Wirkungsweise des Befehls entspricht der von `\hfill`, siehe Kap. 8.2.

Z. B. ergibt der Befehl `A\dotfill_B`

$A \dots\dots\dots B$

Analog dazu gibt es für eine Linie den Befehl `\hrulefill`:

$A \hrulefill B$

Um die Befehle für Text zu verwenden, müssen sie im T_EX-Modus eingegeben werden.

4 Matrizen

Matrizen können über das Fenster **Mathe-Kontrollfläche** oder die **Mathe-Symbolleiste** eingefügt werden. Der Matrix-Knopf in der Symbolleiste erstellt eine 2×2 Matrix deren Spalten- und Zeilenanzahl mit Hilfe der **Tabellen-Symbolleiste** oder des Menüs **Bearbeiten**▷**Zeilen & Spalten** verändert werden kann. Im Fenster **Mathe-Kontrollfläche** werden sie hingegen vor dem Erstellen der Matrix nach der Spalten- und Zeilenanzahl und der Zellausrichtung gefragt. Die vertikale Ausrichtung ist dabei nur bei Matrizen in eingebetteten Formeln von Bedeutung:

Die erste Matrix ist oben
$$\begin{matrix} A & D & G & J \\ B & E & H & K \\ C & F & I & L \end{matrix}$$
 die zweite mittig
$$\begin{matrix} A & D & G & J \\ B & E & H & K \\ C & F & I & L \end{matrix}$$
 und die dritte unten
$$\begin{matrix} A & D & G & J \\ B & E & H & K \\ C & F & I & L \end{matrix}$$
 ausgerichtet.

Die horizontale Ausrichtung gibt an, wie die Spalteneinträge ausgerichtet werden sollen. Dazu wird für jede Spalte ein Buchstabe eingegeben. *l* steht für linksbündig, *c* für mittig und *r* für rechtsbündig. Möchte man z. B. eine 4×4 Matrix erstellen, bei der der Inhalt der ersten Spalte linksbündig, der Inhalt der zweiten und dritten mittig und der Inhalt der letzten rechtsbündig ausgerichtet ist, gibt man für die horizontale Ausrichtung **lccr** an. Normalerweise sind in einer Matrix alle Spalteninhalte zentriert, weswegen die Voreinstellung für jede Spalte ein **c** ist.

Horizontale Ausrichtung

$$\begin{matrix} 10000 & D & G \\ \text{III} : B & 10000 & H \\ C & F & 10000 \end{matrix}, \text{ccc} : \begin{matrix} 10000 & D & G \\ B & 10000 & H \\ C & F & 10000 \end{matrix}, \text{rrr} : \begin{matrix} 10000 & D & G \\ B & 10000 & H \\ C & F & 10000 \end{matrix}$$

Klammern um eine Matrix können entweder mit den Befehlen `\left` und `\right` erstellt werden (Tastenkürzel **Alt-m Klammer**), siehe Kap. 5.1.2, oder man verwendet folgende Befehle:

Befehl	Ergebnis	Befehl	Ergebnis
<code>\bmatrix_2 \times 2 Matrix</code>	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$	<code>\vmatrix_2 \times 2 Matrix</code>	$\begin{vmatrix} 0 & -i \\ i & 0 \end{vmatrix}$
<code>\Bmatrix_2 \times 2 Matrix</code>	$\left\{ \begin{matrix} 0 & -i \\ i & 0 \end{matrix} \right\}$	<code>\Vmatrix_2 \times 2 Matrix</code>	$\left\ \begin{matrix} 0 & -i \\ i & 0 \end{matrix} \right\ $
<code>\pmatrix_2 \times 2 Matrix</code>	$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	<code>\matrix_2 \times 2 Matrix</code>	$\begin{matrix} 0 & -i \\ i & 0 \end{matrix}$

Wurde z. B. `\vmatrix` eingegeben, erscheint ein blaues Kästchen zwischen zwei senkrechten Strichen, in das die Matrix eingefügt wird.

Da alle mehrzeiligen Formeln Matrizen sind, kann man die in Kap. 18.1.3 beschriebene Länge `\arraycolsep` auch zur Änderung des Spaltenabstands von Matrizen verwenden.

Möchte man den Zeilenabstand ändern, verwendet man den Befehl `\arraystretch`. Dieser wird folgendermaßen benutzt:

`\renewcommand{\arraystretch}{Dehnungsfaktor}`

Der Befehl `\renewcommand` weist dabei dem vordefinierten Befehl `\arraystretch` den Dehnungsfaktor zu. Möchte man z. B. den Zeilenabstand verdoppeln, gibt man für den Faktor eine 2 an. Dieser gilt dann für alle folgenden Matrizen. Um wieder zum ursprünglichen Abstand zurückzukehren, weist man `\arraystretch` den Faktor 1 zu.

Um Matrizen in einer Textzeile zu verwenden, benutzt man den Befehl `\smallmatrix`. Gibt man ihn in eine Formel ein, erscheint ein blaues Kästchen mit zwei gestrichelten Linien. In dieses Kästchen wird die Matrix eingegeben. Dies kann nicht über das Fenster **Mathe-Kontrollfläche** geschehen. Man verwendet stattdessen das Menü **Bearbeiten** \triangleright **Zeilen & Spalten** oder die **Mathe-Symbolleiste** um neue Spalten zu erstellen. Neue Zeilen können auch mit **Strg-Enter** erstellt werden.

Dies ist eine Matrix $\begin{pmatrix} A & B \\ C & D \end{pmatrix}$ in einer Textzeile.

Die im Menü **Einfügen** \triangleright **Mathe** verfügbare **Array-Umgebung** erzeugt eine 2×2 Matrix.

5 Klammern und Begrenzungszeichen

5.1 Vertikale Klammern und Begrenzungszeichen

Befehl	Ergebnis
((
{	{
[[
\langle	⟨
\lceil	⌈
\lfloor	⌊
/	/

Befehl	Ergebnis
))
}	}
]]
\rangle	⟩
\rceil	⌈
\rfloor	⌋
\backslash	\
\	

Achtung: Im \TeX -Modus muss für den Backslash der Befehl `\textbackslash` verwendet werden, denn der Befehl `\` steht dort für einen Zeilenumbruch.

Bei allen oben aufgelisteten Zeichen kann die Größe mit den in den folgenden zwei Unterkapiteln beschriebenen Befehlen eingestellt werden. Dabei können statt den Befehlen `\langle` und `\rangle` direkt die Zeichen `<` und `>` verwendet werden.

5.1.1 Manuelle Klammergrößen

Möchte man die Klammergröße manuell festlegen, kann man dies mittels der \LaTeX -Befehle `\big`, `\Big`, `\bigg` und `\Bigg` tun. `\big` steht hierbei für die kleinste und `\Bigg` für die größte Klammer.

Diese Befehle werden benutzt um Klammerebenen optisch hervorzuheben:

\LaTeX stellt folgende Formel so dar: $((A + B)(A - B))^C$

besser sieht das so aus: $\left((A + B)(A - B)\right)^C$

Für die zweite Formel wurde der Befehl `\Big((A+B)(A-B)\Big)^C` verwendet.

Hier alle Klammergrößen in der Übersicht:

`\Bigg(\exp\bigg<\Big[\big\{\ln(3x)\big\}^2\sin(x)\Big]^A\Bigg)^{0,5}`

$$\left(\exp\left\langle\left[\left\{\ln(3x)\right\}^2\sin(x)\right]^A\right\rangle\right)^{0,5}$$

Es gibt außer den `\big`-Befehlen noch die Variante `\bigm`, die etwas mehr Leerraum zwischen Klammer und Klammerinhalt erzeugt und die Variante `\bigl\bigl`, die keinen zusätzlichen Leerraum erzeugt. Das *l* am Ende des Befehls `\bigl` steht für eine öffnende Klammer, für eine schließende Klammer wird dieses durch ein *r* ersetzt. Dabei ist zu beachten, dass eine linke Klammer auch eine schließende Klammer sein darf.

In der folgenden Tabelle sind die Varianten miteinander verglichen:

Befehl	Ergebnis
<code>\Bigm(\bigm(\ln(3x)\bigm)^2\Bigm)</code>	$\left(\left(\ln(3x)\right)^2\right)$
<code>\Big(\big(\ln(3x)\big)^2\Big)</code>	$\left(\left(\ln(3x)\right)^2\right)$
<code>\Bigl(\bigl(\ln(3x)\bigr)^2\Bigr)</code>	$\left(\left(\ln(3x)\right)^2\right)$
<code>\bigl)\ln(3x)\bigr(</code>	$)\ln(3x)($

5.1.2 Automatische Klammergrößen

Klammern für mehrzeilige Ausdrücke können mit den Befehlen `\left` und `\right` oder über das Fenster **Mathe-Kontrollfläche** eingefügt werden. Auf `\left` und `\right` muss unmittelbar die gewünschte Klammer folgen. Die Klammergröße wird dann bei der Ausgabe automatisch berechnet.

normale Klammer: Der Befehl `\ln(\frac{A}{C})` ergibt

$$\ln\left(\frac{A}{C}\right)$$

mehrzeilige Klammer: Der Befehl `\ln\left(\frac{A}{C}\right)` ergibt

$$\ln\left(\frac{A}{C}\right)$$

An Stelle von `\left` und `\right` kann man auch die Tastenkürzel **Alt-m Klammer** verwenden. Das hat den Vorteil, dass man in L^AT_EX sofort die wahre Klammergröße sieht und dass die rechte Klammer gleich mit erstellt wird.

Der Befehl für das letzte Beispiel würde dann lauten: `\ln Alt-m (\frac{A}{C}`

Will man nur eine Klammerhälfte erstellen, schreibt man für die weggelassene Klammer einen Punkt. So ergibt z. B. der Befehl `\left.\frac{A}{B}\right\}`:

$$\left.\frac{A}{B}\right\}$$

Die Befehle `\left` und `\right` werden beim Neu laden des Dokuments von `LyX` automatisch in Klammern der richtigen Größe umgewandelt. Für eine weggelassene Klammer erscheint in `LyX` eine gestrichelte Linie.

Da alle gängigen `LaTeX`-Distributionen `eTeX`, eine Erweiterung von `LaTeX`, verwenden, steht für alle Klammern und Begrenzungszeichen zusätzlich der Befehl `\middle` zur Verfügung. Durch diesen wird das Zeichen an die Höhe der umgebenden Klammern angepasst, was z. B. für physikalische Vektoren von Nutzen ist:

$$\left\langle \phi \middle| J = \frac{3}{2}, M_J \right\rangle$$

Für physikalische Vektoren gibt es ein spezielles `LaTeX`-Paket, das in Kap. 22.4 beschrieben ist.

5.2 Horizontale Klammern

Befehl	Ergebnis
<code>\overbrace{A+B}^3</code>	$\overbrace{A+B}^3$
<code>\underbrace{A+B}_5</code>	$\underbrace{A+B}_5$
<code>\overbrace{\underbrace{A+B}_w}^C</code>	$\overbrace{\underbrace{A+B}_w}^C$

Beim letzten Beispiel spielt es keine Rolle, ob man zuerst `\overbrace` oder `\underbrace` eingibt.

Benötigt man Klammern, die sich überschneiden, muss man die in Kap. 18 beschriebenen mehrzeiligen Formeln verwenden.

$$A = \underbrace{\underbrace{ggg + bbq}_r + ddd}_s$$

In die erste Zeile wird die Formel zusammen mit der ersten Klammer eingegeben. Dabei ist es wichtig, dass vor dem ersten *d* der Abstandsbehl `\:` eingegeben wird, siehe Kap. 8.1. Denn die nach dem *q* endende Klammer verhindert, dass das nachfolgende *+* von Leerraum umgeben ist, weil eine Klammer nicht als Zeichen gilt, siehe Kap. 10.3. In die zweite Zeile wird die zweite Klammer eingegeben. Da sie vor dem *b* beginnen soll, gibt man zuerst `\hphantom{ggg+ \:}` ein⁹. Der Leerraum wird

⁹mehr zu `\hphantom` siehe Kap. 3.7.

benötigt, weil das $+$ in der Formel von Leerraum umgeben ist. Die Klammer wird unter den Befehl `\hphantom{bbqq+dddd}` gesetzt.

Komplizierter wird es, wenn sich Klammern über und unter der Formel überschneiden, wie in folgendem Beispiel:

$$A = \underbrace{gggg}_r + \overbrace{bbqq}_s + dddd$$

Die erste Formelzeile ist identisch mit der zweiten Zeile des vorigen Beispiels, mit dem Unterschied, dass sich die Klammer oben befindet. Die zweite Zeile enthält die Formel zusammen mit der zweiten Klammer. Damit sich zwischen oberer Klammer in der ersten Zeile und der Formel in der zweiten Zeile kein Leerraum befindet, muss der Zeilenabstand verändert werden. Aufgrund eines Fehlers in L^AT_EX ist dies nicht ohne Weiteres möglich. Es muss als Lösung des Problems der globale Formelzeilenabstand `\jot` mit folgendem Befehl im T_EX-Modus geändert: `\setlength{\jot}{-6pt}` und anschließend wieder auf den Standardwert zurückgesetzt werden. Genauer zum Zeilenabstand in Formeln ist in Kap. 18.1.2 erklärt.

6 Pfeile

6.1 Horizontale Pfeile

Befehl	Ergebnis
<code>\gets</code>	\leftarrow
<code>\Leftarrow</code>	\Leftrightarrow
<code>\longleftarrow</code>	\longleftarrow
<code>\Longleftarrow</code>	\Longleftarrow
<code>\leftharpoonup</code>	\leftharpoonup
<code>\leftharpoondown</code>	\leftharpoondown
<code>\hookleftarrow</code>	\hookleftarrow

Befehl	Ergebnis
<code>\leftrightarrow</code>	\leftrightarrow
<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\longleftrightarrow</code>	\longleftrightarrow
<code>\Longleftrightarrow</code>	\Longleftrightarrow
<code>\rightleftharpoons</code>	\rightleftharpoons

Befehl	Ergebnis
<code>\to</code>	\rightarrow
<code>\Rightarrow</code>	\Rightarrow
<code>\longrightarrow</code>	\longrightarrow
<code>\Longrightarrow</code>	\Longrightarrow
<code>\rightharpoonup</code>	\rightharpoonup
<code>\rightharpoondown</code>	\rightharpoondown
<code>\hookrightarrow</code>	\hookrightarrow

Befehl	Ergebnis
<code>\mapsto</code>	\mapsto
<code>\longmapsto</code>	\longmapsto
<code>\leadsto</code>	\leadsto
<code>\dashrightarrow</code>	\dashrightarrow

Als Akzent verwendete Pfeile, wie z. B. Vektorpfeile, sind in Kap. 7 aufgelistet.

Des Weiteren gibt es die beschriftbaren Pfeile `\xleftarrow` und `\xrightarrow`. Gibt man einen dieser Befehle in eine Formel ein, erscheint ein Pfeil mit zwei blauen Kästchen, in die man die Beschriftung eingeben kann. Die Pfeillänge passt sich automatisch der Beschriftung an.

Befehl	Ergebnis
$F(a)\backslash xleftarrow{x=a}\downarrow x>0\rightarrow F(x)$	$F(a) \xleftarrow[x>0]{x=a} F(x)$
$F(x)\backslash xrightarrow{x=a}\downarrow x>0\rightarrow F(a)$	$F(x) \xrightarrow[x>0]{x=a} F(a)$

6.2 Vertikale und diagonale Pfeile

Befehl	Ergebnis
$\backslash uparrow$	\uparrow
$\backslash Uparrow$	\Uparrow
$\backslash updownarrow$	\updownarrow
$\backslash Updownarrow$	\Updownarrow
$\backslash Downarrow$	\Downarrow
$\backslash downarrow$	\downarrow

Befehl	Ergebnis
$\backslash nearrow$	\nearrow
$\backslash searrow$	\searrow
$\backslash swarrow$	\swarrow
$\backslash nwarrow$	\nwarrow

Vertikale Pfeile können auch als Begrenzungszeichen zusammen mit den in Kap. 5.1.1 und 5.1.2 beschriebenen Befehlen verwendet werden.

7 Akzente

7.1 Akzente für ein Zeichen¹⁰

Befehl	Ergebnis
$\backslash dot_A$	\dot{A}
$\backslash ddot_A$	\ddot{A}
$\backslash dddot_A$	\dddot{A}
$\backslash dddd\dot{A}$	\ddddot{A}
$\backslash vec_A$	\vec{A}
$\backslash bar_A$	\bar{A}
$\backslash mathring_A$	\mathring{A}

Befehl	Ergebnis
$\backslash tilde_A$	\tilde{A}
$\backslash hat_A$	\hat{A}
$\backslash check_A$	\check{A}
$\backslash acute_A$	\acute{A}
$\backslash grave_A$	\grave{A}
$\backslash breve_A$	\breve{A}

Wenn man in einer Formel Umlaute verwenden will, schreibt man ein Anführungszeichen vor den Selbstlaut. Diese beiden Zeichen bilden dann für \LaTeX ein Zeichen und müssen immer unmittelbar aufeinander folgen. Man beachte, dass im Gegensatz zu $\backslash ddot$ „echte“ Umlaute entstehen, was folgendes Beispiel zeigt:

¹⁰Akzente im Text siehe Kap. 16.2.

Befehl	Ergebnis
"i	<i>i</i>
\ddot{u}	\ddot{u}

Ein weiterer Vorteil gegenüber `\ddot` ist, dass man Umlaute direkt in mathematischen Text konvertieren kann. Denn die obigen Akzentbefehle sind *im mathematischen Textmodus nicht erlaubt*. Um ein akzentuiertes Zeichen in mathematischen Text zu überführen, darf nur das Zeichen unter dem Akzent umgewandelt werden. Das gilt auch für alle anderen Umwandlungen wie z. B. in kursiv oder fett.

Im mathematischen Textmodus können die Umlaute ä, ö und ü direkt eingegeben werden.

7.2 Akzente für Operatoren

Zum Akzentuieren von Operatoren gibt es die Befehle `\overset` und `\underset`, mit denen man Zeichen über bzw. unter einen Operator setzen kann. Mit `\sideset` können Zeichen vor und nach einen Operator gesetzt werden. Das Befehlsschema lautet:

`\sideset{Zeichen davor}{Zeichen dahinter}`

`\sideset` müssen immer vor dem zu akzentuierenden Operator stehen. Es kann auch mit mehreren Zeichen und sogar mit anderen Operatoren und Symbolen akzentuiert werden. Will man mit `\sideset` z. B. nur Zeichen hinter einen Operator setzen, schreibt man nichts zwischen die ersten geschweiften Klammern, die Klammern dürfen aber nicht weggelassen werden.

Z. B. ergibt der Befehl `\sideset{\rightarrow\{'\rightarrow\sum_{k=1}^n` :

$$\sum_{k=1}^n$$

Der Befehl `\overset{\mathfrak{a}}` ergibt:

$$\overset{\mathfrak{a}}{\otimes}$$

Wie man am letzten Beispiel sieht, kann man mit `\overset` bzw. `\underset` außer Operatoren auch Symbole und Zeichen akzentuieren. Mit `\sideset` ist dies nicht möglich.

7.3 Akzente für mehrere Zeichen

Befehl	Ergebnis	Befehl	Ergebnis
<code>\overleftarrow{A=B}</code>	$\overleftarrow{A=B}$	<code>\overrightarrow{A=B}</code>	$\overrightarrow{A=B}$
<code>\underleftarrow{A=B}</code>	$\underleftarrow{A=B}$	<code>\underrightarrow{A=B}</code>	$\underrightarrow{A=B}$
<code>\overleftrightarrow{A=B}</code>	$\overleftrightarrow{A=B}$	<code>\widetilde{A=B}</code>	$\widetilde{A=B}$
<code>\underleftrightarrow{A=B}</code>	$\underleftrightarrow{A=B}$	<code>\widehat{A=B}</code>	$\widehat{A=B}$

Mit diesen Befehlen können beliebig viele Zeichen akzentuiert werden. Die Akzente `\widetilde` und `\widehat` werden aber im Ausdruck nur mit einer Länge von etwa drei Zeichen gesetzt, was folgendes Beispiel verdeutlicht:

$$A + \widetilde{B} = C - D$$

Mit den im vorigen Kapitel beschriebenen Befehlen `\overset` und `\underset` ist es auch möglich mehrere Zeichen zu akzentuieren. Der Befehl `\underset{A=B}{***}` ergibt:

$$\underset{***}{A=B}$$

8 Leerraum

8.1 Vordefinierter Leerraum

Ab und an ist es notwendig in einer Formel Leerraum einzufügen. Dazu gibt man ein geschütztes Leerzeichen ein. Drückt man nun die Leertaste mehrmals hintereinander, kann man aus acht verschiedenen Größen wählen. Man kann die Leerräume auch mit einzelnen Befehlen aufrufen. Unabhängig vom eingegebenen Befehl kann man durch Drücken der Leertaste wieder aus allen Größen wählen.

Befehl	<code>\,</code>	<code>\:</code>	<code>\;</code>	<code>\quad</code>	<code>\qquad</code>	<code>\!</code>
Anzahl der Leertastendrucke nach Eingabe des geschützten Leerzeichens	0	1	2	3	4	5
Ergebnis	$A B$	$A B$	$A B$	$A B$	$A B$	AB

Die letzte Größe scheint überhaupt keinen Leerraum zu produzieren. In LyX ist diese allerdings auch im Gegensatz zu allen anderen rot eingefärbt, denn sie erstellt negativen Leerraum. Es gibt noch zwei weitere negative Leerräume:

Befehl	<code>\negmedspace</code>	<code>\negthickspace</code>
Anzahl der Leertastendrucke nach Eingabe des geschützten Leerzeichens	6	7
Ergebnis	AB	AB

Negative Leerräume bewirken, dass sich benachbarte Zeichen überschneiden. Man kann auf diese Weise Ligaturen erzwingen, was z. B. für Summenzeichen von Nutzen ist.

Befehl	Ergebnis
<code>\sum\sum_f_{kl}</code>	$\sum \sum f_{kl}$
<code>\sum\negmedspace\sum_f_{kl}</code>	$\sum \sum f_{kl}$

Relationen, wie z. B. Gleichheitszeichen, sind immer von Leerraum umgeben. Will man dies unterdrücken, schreibt man das Gleichheitszeichen in eine \LaTeX -Klammer. Die Wirkung kann man an folgendem Beispiel gut erkennen:

normale Gleichung	$A = B$
Gleichung ohne Leerraum	$A=B$

Der Befehl für die letzte Formel lautet: $\mathbf{A}\{\mathbf{=}\mathbf{\rightarrow}\mathbf{B}$

Eine häufige Anwendung von Leerräumen sind physikalische Einheiten, denn der allgemein übliche Leerraum zwischen Wert und Einheit ist kleiner als ein Leerzeichen. Man verwendet daher einen der beiden kleinsten positiven Leerräume. Bei Einheiten im Text fügt man zwischen Wert und Einheit statt des Leerzeichens den kleinsten Leerraum über das Menü **Einfügen** \triangleright **Besondere Formatierung** \triangleright **Kleiner Abstand** (Tastenkürzel **Strg-Shift-Leertaste**) ein.

Ein Beispiel zur Veranschaulichung:

24 kW·h	Leerzeichen zwischen Wert und Einheit
24 kW·h	kleinster Leerraum zwischen Wert und Einheit
24 kW·h	zweit kleinster Leerraum zwischen Wert und Einheit

8.2 Variabler Leerraum¹¹

Leerraum mit vorgegebener Länge lässt sich in \LaTeX mit dem Befehl `\hspace{Länge}` einfügen. Dabei können alle Längeneinheiten verwendet werden, die am Ende von Kap. 2 aufgelistet sind, mit Ausnahme der Einheit „mu“. Die Länge darf auch negativ sein. Um so viel Leerraum einzufügen, dass die Formel allen verfügbaren Platz einnimmt, benutzt man den Befehl `\hfill`.

¹¹vertikaler Leerraum siehe Kap. 22.16

Befehl	Ergebnis	
$A=B\hspace{3cm}\rightarrow A\neq C$	$A = B$	$A \neq C$
$A\hspace{-1mm}\rightarrow A\neq A$	$AA \neq A$	
$A=A\hfill B=B$	$A = A$	$B = B$

Im letzten Beispiel ist der verfügbare Platz durch die Länge des längsten Spalten-
eintrags der Tabelle vorgegeben. In einer eingebetteten Formel ist der Platz von der
Länge der Zeile abhängig, in die `\hfill` eingegeben wird. Füllt eine Zeile die Seiten-
breite komplett aus, wird demnach kein Leerraum erstellt. `\hfill` wirkt in abgesetz-
ten Formeln nur, wenn der Formelstil **Halblinks** eingestellt ist (Formelstile siehe
Kap. 17).

Die Befehle `\hspace` und `\hfill` können auch im \TeX -Modus eingegeben werden, um
sie für Text zu verwenden:

Dies ist eine Zeile mit $\hspace{2cm}$ 2 cm Leerraum.

Dies ist eine Zeile mit \hfill maximalem Leerraum.

Im letzten Beispiel wurde `\hfill` über das Menü **Einfügen** \triangleright **Besondere Forma-
tierung** \triangleright **Variabler horiz. Abstand** eingefügt. Das hat den Vorteil, dass man die
Größe des Leerraums bereits in \LaTeX sieht.

Befinden sich die Befehle am Anfang einer Zeile, jedoch nicht am Anfang eines Ab-
satzes, werden sie von \LaTeX ignoriert. Um das zu verhindern, verwendet man statt
`\hspace` den Befehl `\hspace*`. Bei `\hfill` fügt man vor dem Befehl einen geschütz-
ten Zeilenumbruch und im \TeX -Modus mit dem Befehl `\vphantom{ }` einen leeren
Platzhalter¹² ein.

8.3 Leerraum neben eingebetteten Formeln

Über die Länge `\mathsurround` lässt sich der Leerraum einstellen, der neben einge-
bettete Formeln gesetzt wird. Der Wert einer Länge wird mit dem Befehl `\setlength`
gesetzt. Dieser besitzt folgendes Befehlsschema:

`\setlength{Längenname}{Wert}`

¹²Platzhalter siehe Kap. 3.7

Um z. B. `\mathsurround` den Wert 5 mm zuzuweisen, gibt man im T_EX-Modus den Befehl

```
\setlength{\mathsurround}{5mm}
```

ein. Neben alle eingebetteten Formeln, die auf den Befehl folgen, wird nun 5 mm Leerraum gesetzt:

Dies ist eine Zeile in der sich $A = B$ eine Formel mit umgebendem Leerraum befindet.

Um wieder zum voreingestellten Wert zurückzukehren, setzt man `\mathsurround` auf den Wert 0 pt.

9 Boxen und Rahmen¹³

9.1 Boxen mit Rahmen

Es ist möglich Formeln oder Teile davon umrandet darzustellen. Dazu gibt es die Befehle `\fbox` und `\boxed`.

Gibt man einen dieser Befehle in eine Formel ein, erscheint ein Eingabefeld in einem Rahmen. Bei `\fbox` muss in diesem Eingabefeld mit **Strg-m** noch eine Formel erstellt werden¹⁴, denn der Inhalt dieser Box wird sonst als mathematischer Text behandelt. Verwendet man `\boxed`, wird automatisch im Rahmen eine neue Formel erzeugt.

Der Befehl `\fbox` ist nur schlecht zur Umrandung von abgesetzten Formeln geeignet, denn die Formel wird dann immer in Textzeilengröße gesetzt. Im Gegensatz dazu ist `\boxed` nicht zur Umrandung von Formeln in einer Textzeile geeignet, denn die Formel wird dann immer in der Größe einer abgesetzten Formel gesetzt.

Als Erweiterung zu `\fbox` gibt es den Befehl `\framebox`, bei dem man zusätzlich die Rahmenbreite und die Ausrichtung einstellen kann. `\framebox` wird nach folgendem Schema verwendet:

```
\framebox[Rahmenbreite][Position]{Boxinhalt}
```

Die Position kann entweder *l* oder *r* sein. *l* bewirkt linksbündige, *r* rechtsbündige Ausrichtung der Formel in der Box. Gibt man keine Position an, wird die Formel zentriert.

Gibt man keine Breite an, darf auch keine Position angegeben werden. In diesem Falle wird die Rahmenbreite wie bei `\fbox` an den Boxinhalt angepasst.

Wenn man den Befehl `\framebox` eingibt, erscheint eine Box mit drei Eingabefeldern. Die ersten beiden sind von eckigen Klammern umgeben und stehen für die

¹³Weitere Boxen sind in Kap. 22.15 beschrieben.

¹⁴Durch einen Fehler in L_yX ist es momentan nicht möglich mit **Strg-m** eine neue Formel zu erzeugen, siehe [L_yX-Fehlernummer #1435](#)

beiden optionalen Argumente. In das dritte Feld wird wie bei `\fbox` die Formel eingegeben.

Befehl ^a	Ergebnis
<code>\fbox{Strg-m \int A=B}</code>	$\int A = B$
<code>\boxed{\int A=B}</code>	$\int A = B$
<code>A+\fbox{B}</code>	$A + \boxed{B}$
<code>\framebox{20mm}\rightarrow\rightarrow\text{Strg-m \frac{A}{B}}</code>	$\frac{A}{B}$

^aDurch einen Fehler in L^AT_EX ist es momentan nicht möglich mit **Strg-m** eine neue Formel zu erzeugen, siehe [L^AT_EX-Fehlernummer #1435](#)

Zusätzlich kann man die Rahmendicke einstellen. Dazu müssen vor der Formel im T_EX-Modus die Befehle

`\fboxrule „Dicke“ \fboxsep „Abstand“`

eingegeben werden. „Abstand“ legt dabei den minimalen Abstand zwischen Rahmen und dem ersten Zeichen in der Box fest. Nehmen wir z. B. folgende umrandete Formel:

$$\boxed{A + B = C}$$

Vor dieser Formel wurde im T_EX-Modus

`\fboxrule 2mm \fboxsep 3mm`

eingegeben. Diese eingegebenen Werte behalten für alle folgenden Boxen ihre Gültigkeit.

Möchte man wieder zur Ausgangsrahmengröße zurückkehren, gibt man vor der nächsten Formel im T_EX-Modus folgende Befehle ein:

`\fboxrule 0.4pt \fboxsep 3pt`

Es kann auch Text umrandet werden. Dabei ist zu beachten, dass eine Box nur eine Zeile enthalten kann. Boxen für mehrere Zeilen sind in Kap. 9.4 beschrieben. Dort ist außerdem erklärt, wie man Formeln inklusive der Formelnummer umrandet.

Um Text zu umranden, schreibt man vor das betreffende Textstück im T_EX-Modus z. B. den Befehl `\framebox[5cm]{`. Nach dem Textstück schließt man die Box, indem man im T_EX-Modus ein `}` eingibt. Die Befehle `\fboxrule` und `\fboxsep` können wie oben beschrieben verwendet werden.

Dies ist eine Zeile mit $\boxed{\text{zwei umrandeten}}$ Wörtern.

9.2 Boxen ohne Rahmen

Für Boxen ohne Rahmen gibt es folgende Box-Befehle: `\mbox`, `\makebox` und `\raisebox`

Mit `\raisebox` kann man eine Box hoch- bzw. tiefstellen. Allerdings behalten die Zeichen in der Box im Gegensatz zum normalen Hochstellen die Schriftgröße bei. `\raisebox` wird dabei mit folgendem Schema benutzt:

`\raisebox{Höhe}{Boxinhalt}`

Soll in der Box eine Formel stehen, muss man wie bei `\fbox` eine weitere Formel erstellen.

Befehl	Ergebnis
<code>H\raisebox{2mm}{\{al\rightarrow lo}</code>	$H^{\text{al}}l_o$
<code>H\raisebox{-2mm}{\{al\rightarrow lo}</code>	$H_{\text{al}}l_o$
<code>A=\raisebox{-2mm}{\{\Strg-m \sqrt{B}</code>	$A = \sqrt{B}$

Die letzte Formel muss derzeit komplett im \TeX -Modus erstellt werden, da \LyX statt der weiteren Formel eine Box einfügt¹⁵.

Der Befehl `\mbox` ist äquivalent zu `\fbox` und `\makebox` ist äquivalent zu `\framebox`, nur dass es keinen Rahmen gibt (Rahmendicke = 0).

Wie man mit `\mbox` Zeichen vor einem Zeilenumbruch schützt, ist in Kap. 22.17.1 beschrieben.

9.3 Farbige Boxen

Damit die in diesem Kapitel angegebenen Befehle benutzt werden können, muss im \LaTeX -Vorspann das \LaTeX -Paket **color**¹⁶ mit der Zeile

`\usepackage{color}`

geladen werden¹⁷.

Um Boxen einzufärben, verwendet man den Befehl `\colorbox`. Dieser wird nach folgendem Schema benutzt:

`\colorbox{Farbe}{Boxinhalt}`

¹⁵siehe [LyX-Fehlernummer #1435](#)


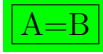

¹⁶Das Paket **color** ist Teil jeder \LaTeX -Standardinstallation.

¹⁷Wird irgendwo im Dokument Text mit einer vordefinierten Farbe gefärbt, lädt \LyX das Paket **color** automatisch. Dadurch kann es vorkommen, dass das Paket doppelt geladen wird. Dies ruft aber keine Fehler hervor.

Der Boxinhalt kann auch wieder eine Box sein und eine `\colorbox` kann auch ein Teil einer anderen Box sein (siehe 2. und 3. Beispiel). Soll in der Box eine Formel stehen, muss wie bei `\fbox` eine weitere Formel erstellt werden.¹⁸

Aus folgenden vordefinierten Farben kann gewählt werden:

black, blue, cyan, green, magenta, red, white und **yellow**

Befehl	Ergebnis
<code>\colorbox{yellow}\{A=B</code>	
<code>\colorbox{green}\{\fbox{A=B}</code>	
<code>\fbox{\colorbox{green}\{A=B</code>	

`\colorbox` färbt nur die Box, nicht aber die Zeichen in der Box ein. Um alle Zeichen zu färben, markiert man die ganze Formel und wählt im Menü **Bearbeiten** > **Textstil** die gewünschte Farbe aus. Die Formelnummer hat dann dieselbe Farbe wie die Formel. Soll die Formelnummer eine andere Farbe als die Zeichen haben, muss innerhalb der Formel die Farbe geändert werden.

Ein Beispiel:

$$\int A = B \tag{1}$$

$$\int A = B \tag{2}$$

Formel (1) ist komplett rot gefärbt.

Formel (2) wurde zuerst komplett grün gefärbt, um die Farbe der Formelnummer festzulegen. Anschließend wurden die Formelzeichen rot gefärbt.

Möchte man den Rahmen anders als den Rest der Box färben, verwendet man den Befehl `\fcolorbox` nach folgendem Schema:

`\fcolorbox{Rahmenfarbe}{Farbe}{Boxinhalt}`

`\fcolorbox` stellt also eine Erweiterung des Befehls `\colorbox` dar. Die Rahmenbreite wird, wie bei `\framebox`, mit `\fboxrule` und `\fboxsep` festgelegt. Dazu ein Beispiel:



¹⁸Das gilt auch für den Befehl `\fcolorbox`. Dabei ist folgender L^AT_EX-Fehler zu beachten:
[L^AT_EX-Fehlernummer #1435](#)

Diese Formel wurde mit dem Befehl `\fcolorbox{cyan}{\magenta}{A=B}` erstellt.

Um andere als die vordefinierten Farben zu verwenden, muss man diese selbst definieren.

Man kann z. B. mit folgender L^AT_EX-Vorspannzeile die Farbe „dunkelgruen“¹⁹ definieren:

```
\definecolor{dunkelgruen}{cmyk}{0.5, 0, 1, 0.5}
```

cmyk ist der Farbraum und steht für die Farben **c**yan, **m**agenta, **y**ellow und **b**lack. Die vier Zahlen geben der Reihe nach den Anteil der vier Farben an. Für den Anteil kann man Werte im Bereich 0 - 1 angeben. Anstelle von **cmyk** kann man zur Definition auch den Farbraum **rgb** verwenden. **rgb** steht für **r**ed, **g**reen und **b**lue, so dass man dann mit drei Zahlen die Anteile dieser Farben festlegen kann. Des Weiteren gibt es noch den Farbraum **gray**, bei dem man mit einer Zahl den Graustufenanteil festlegen kann.

Als Beispiel eine gerahmte Box mit der neu definierten Farbe **darkgreen**, bei der außerdem die Zeichenfarbe **yellow** gewählt wurde:

$$\int A \, dx = \frac{\sqrt[5]{B}}{\ln\left(\frac{1}{3}\right)} \quad (3)$$

Selbst definierte Farben können mit dem Befehl `\textcolor` auch für Text verwendet werden:

Dieser Satz ist „dunkelgruen“.

`\textcolor` wird nach dem Schema `\textcolor{Farbe}{zu färbende Zeichen}` verwendet.

9.4 Absatzboxen

Über das Menü **Einfügen** \triangleright **Box** kann man eine Box erstellen, die einen eigenen Absatz darstellt, eine sogenannte *Parbox* (Absatzbox).

Das folgende Beispiel zeigt eine Absatzbox in einer Zeile, die zusätzlich umrahmt ist:

Dies ist eine Zeile

Dies ist die Absatzbox. Sie ist genau 5 cm lang und kann auch Formeln enthalten: $\int A \, ds = C$

 mit einer Absatzbox.

¹⁹In Befehlen darf kein Umlaut stehen, siehe Kap. 20.

Solch eine Box wird erstellt, indem man mit der rechten Maustaste auf die graue Box klickt, die eingefügt wird wenn man das Menü **Einfügen** **Box** aufruft. Es erscheint dann ein Fenster in dem man die Boxeigenschaften festlegt. In diesem Fall: *Typ*: rechteckige Box, *innere Box*: Absatzbox, *Breite*: 5 cm, *vertikale Boxausrichtung*: mittig

In L^AT_EX wird eine Absatzbox mit dem Befehl `\parbox` erstellt, welcher folgendes Schema besitzt:

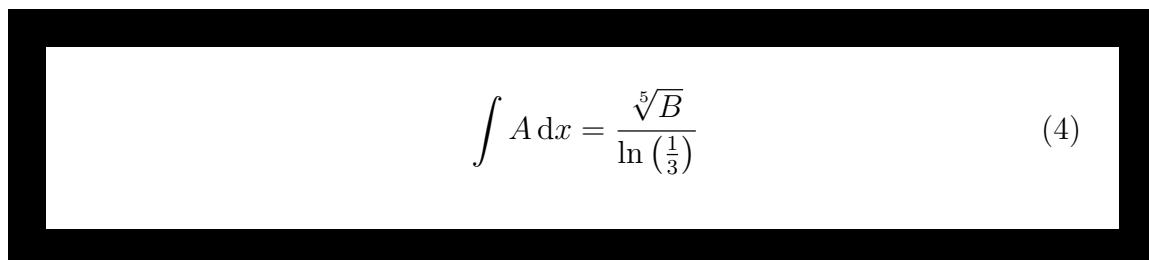
`\parbox[Position]{Breite}{Boxinhalt}`

Die Positionen *b* und *t* sind möglich. *b* steht für bottom und bedeutet, dass die letzte Absatzzeile in den umstehenden Text eingepasst wird. Bei *t* wie top geschieht dies für die erste Zeile. Wird keine Position angegeben, wird der Absatz in mittlerer Höhe eingepasst.

Möchte man Formeln komplett mit Formelnummer umranden, muss man die Formel in eine Absatzbox setzen. Dazu schreibt man im T_EX-Modus vor die Formel den Befehl `\fbox{\parbox{\linewidth-2\fboxsep-2\fboxrule}{`. Hierbei steht `\linewidth` für die im Dokument eingestellte Zeilenbreite. Da sich der Rahmen außerhalb der Absatzbox befindet, muss man von der Zeilenbreite zweimal die Rahmendicke und den Rahmenabstand abziehen. Damit man in Argumenten multiplizieren und subtrahieren kann, muss im L^AT_EX-Vorspann das Paket **calc**²⁰ mit der Zeile

`\usepackage{calc}`

geladen werden. Nach der Formel werden beide Boxen geschlossen, indem man `}}` im T_EX-Modus eingibt. Dazu ein Beispiel:



$$\int A \, dx = \frac{\sqrt[5]{B}}{\ln\left(\frac{1}{3}\right)} \quad (4)$$

Da im Argument von `\fbox` eine Absatzbox verwendet wird, braucht man nicht die in Kap. 9.1 beschriebene weitere Formel einzufügen. Aus dem selben Grund gibt es in diesem Fall bei der Formelgröße keinen Unterschied zwischen `\fbox` und `\boxed`.

Sehr nützlich sind Absatzboxen, wenn man Formeln kommentieren möchte. Dazu verwendet man `\parbox` in Verbindung mit dem Befehl `\tag`. (mehr zu `\tag` siehe Kap. 19.4)

Ein Beispiel einer mit Hilfe von `\parbox` kommentierten Formel:

²⁰`calc` ist Teil jeder L^AT_EX-Standardinstallation.

$$5x - 7b = 3b$$

Dies ist eine Beschreibung.
Sie ist deutlich von der Formel getrennt und mehrzeilig.

So eine Formel muss komplett im $\text{T}_{\text{E}}\text{X}$ -Modus eingegeben werden, da $\text{L}_{\text{A}}\text{X}$ den Befehl `\parbox` in einer Formel noch nicht unterstützt. Die Formel wird mit folgender Befehlssequenz erstellt:

Der Befehl `\[5x-7b=3b\tag*\{\parbox{5cm}\}` wird im $\text{T}_{\text{E}}\text{X}$ -Modus eingegeben²¹. Dann folgt die Beschreibung als normaler Text und zuletzt gibt man `]\]` im $\text{T}_{\text{E}}\text{X}$ -Modus ein. Die Befehle `\[` und `\]` erstellen hierbei eine abgesetzte Formel.

Die Vorteile von `\parbox` erkennt man an folgendem Beispiel, in dem mit Hilfe des mathematischen Textmodus „kommentiert“ wurde:

$5x-7b=3b$ Dies ist eine Beschreibung. Leider ist sie nicht von der Formel getrennt ...

10 Operatoren

10.1 Große Operatoren

Befehl	Ergebnis
<code>\int</code>	\int
<code>\intop</code>	\intop
<code>\oint</code>	\oint
<code>\ointop</code>	\ointop
<code>\prod</code>	\prod
<code>\coprod</code>	\coprod
<code>\bigcap</code>	\bigcap
<code>\bigcup</code>	\bigcup

Befehl	Ergebnis
<code>\sum</code>	\sum
<code>\bigodot</code>	\bigodot
<code>\bigotimes</code>	\bigotimes
<code>\bigoplus</code>	\bigoplus
<code>\bigwedge</code>	\bigwedge
<code>\bigvee</code>	\bigvee
<code>\bigsqcup</code>	\bigsqcup
<code>\biguplus</code>	\biguplus

Die Operatoren werden als groß bezeichnet, da sie größer sind als die zum Teil gleich aussehenden binären Operatoren. Zu allen großen Operatoren kann man Grenzen angeben, wie im nächsten Abschnitt beschrieben.

Die Operatoren `\intop` und `\ointop` unterscheiden sich von `\int` bzw. `\oint` in der Art wie die Operatorgrenzen dargestellt werden, siehe Kap. 10.2.

²¹Ist der Formelstil **Halblinks** aktiviert, kann `\tag*\{` auch durch `\hfill` ersetzt werden. (Formelstile siehe Kap. 17)

Hinweise zu Integralen

Der Buchstabe d in einer Integralgleichung ist ein Operator, der deswegen aufrecht erscheinen sollte. Dazu markiert man das d und wandelt es mit der Tastenkombination **Alt-z r**²² um. Anschließend fügt man vor dem d , wie bei Operatoren üblich, den kleinsten Leerraum ein.

Für Mehrfachintegrale gibt es folgende Befehle:

Befehl	Ergebnis	Befehl	Ergebnis
<code>\iint</code>	\iint	<code>\iiint</code>	\iiint
<code>\iiint</code>	\iiint	<code>\idotsint</code>	$\int \cdots \int$

10.2 Grenzen

Grenzen werden durch Hoch- bzw. Tiefstellen erzeugt:

Befehl	Ergebnis
<code>\int^{\infty}_{-\infty} A(x) dx</code>	$\int_0^{\infty} A(x) dx$
<code>\prod^{\infty}_{-\infty} A(x)</code>	$\prod_0^{\infty} A(x)$

Bei einer Formel in einer Textzeile werden die Grenzen rechts neben den Operator gesetzt. In einer abgesetzten Formel werden, außer bei Integralen, die Grenzen über bzw. unter den Operator gesetzt.

Um zu erzwingen, dass die Grenzen neben den Operator gesetzt werden, setzt man den Cursor direkt hinter den Operator und setzt über das Menü **Bearbeiten** \triangleright **Mathe** \triangleright **Art der Operatorgrenzen ändern** die Grenzen auf **Eingebettet** (Tastenkombination **Alt-m l**). Dazu folgendes Beispiel:

Die normale abgesetzte Formel sieht so aus:

$$\sum_{x=0}^{\infty} \frac{1}{x^2}$$

So sieht die Formel aus, wenn man die Grenzen neben das Summenzeichen setzt:

$$\sum_{x=0}^{\infty} \frac{1}{x^2}$$

Bei Integralen werden mit Ausnahme von `\intop` und `\ointop` die Grenzen standardmäßig neben den Operator gesetzt. Bei Mehrfachintegralen werden die Grenzen

²²Schriftstile siehe Kap. 11.1

jedoch meistens unter den Operator gesetzt. Im folgenden Beispiel wurde die Grenze deswegen auf **Anzeige** gesetzt und damit unter die Integrale geschrieben:

$$\iiint_V X \, dV = U \quad (5)$$

Um Bedingungen für Grenzen anzugeben, gibt es die Befehle `\subarray` und `\substack`. Um z. B. diesen Ausdruck

$$\sum_{\substack{0 < k < 1000 \\ k \in \mathbb{N}}}^n k^{-2} \quad (6)$$

zu erstellen, muss man Folgendes tun:

Zuerst gibt man den Befehl `\sum^n_{\sub}` ein. Man befindet sich nun in einem blauen Kästchen unter dem Summenzeichen. In dieses gibt man den Befehl `\subarray_{\sub}` ein. Rein visuell geschieht gar nichts, man kann jetzt aber mehrere Zeilen untereinander schreiben. Mit Eingabe des erzwungenen Zeilenumbruchs (**Strg-Enter**) wird eine neue Zeile erstellt. Gibt man `0 < k < 1000` **Strg-Enter** ein, erscheint ein Kästchen für eine zweite Zeile.

Die Ausrichtung der Zeilen kann über die **Tabellen-Symbolleiste** oder das Menü **Bearbeiten ▸ Zeilen & Spalten** eingestellt werden.

Der Befehl `\substack` ist äquivalent zu `\subarray`, mit dem Unterschied dass die Zeilen immer zentriert ausgerichtet sind.

Wie bei Formel (6) kann es vorkommen, dass sich neben dem Operator zu viel Leerraum befindet, denn das dem Operator folgende Zeichen wird neben die Grenzen gesetzt.

Um das zu verhindern, kann man folgendes Makro in den L^AT_EX-Vorspann einfügen:

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

Dadurch wird der Befehl `\mathclap` definiert, der die Breite der Grenze auf 0 pt setzt. Das Befehlsschema lautet

```
\mathclap{Grenze}
```

wobei die Grenze auch aus mehreren Bedingungen bestehen kann.

Auf Formel (6) angewandt, gibt man also den Befehl

```
\sum_{\mathclap{\substack{0 < k < 1000}}} \text{Strg-Enter}
```


ein, um die untere Grenze zu erstellen. Der Summand befindet sich nun direkt hinter dem Summenzeichen:

$$\sum_{\substack{0 < k < 1000 \\ k \in \mathbb{N}}}^n k^{-2}$$

In Kap. 10.4 ist beschrieben, wie man eine Grenze für mehrere Operatoren verwenden kann.

10.3 Binäre Operatoren

Binäre Operatoren sind von Leerraum umgeben, wenn sich vor und hinter ihnen ein Zeichen befindet.

Befehl	Ergebnis	Befehl	Ergebnis	Befehl	Ergebnis
<code>+</code>	$+$	<code>\nabla</code>	∇	<code>\oplus</code>	\oplus
<code>-</code>	$-$	<code>\bigtriangledown</code>	\bigtriangledown	<code>\ominus</code>	\ominus
<code>\pm</code>	\pm	<code>\bigtriangleup</code>	\bigtriangleup	<code>\otimes</code>	\otimes
<code>\mp</code>	\mp	<code>\Box</code>	\square	<code>\oslash</code>	\oslash
<code>\cdot</code>	\cdot	<code>\cap</code>	\cap	<code>\odot</code>	\odot
<code>\times</code>	\times	<code>\cup</code>	\cup	<code>\amalg</code>	\amalg
<code>\div</code>	\div	<code>\dagger</code>	\dagger	<code>\uplus</code>	\uplus
<code>*</code>	$*$	<code>\ddagger</code>	\ddagger	<code>\setminus</code>	\setminus
<code>\star</code>	\star	<code>\wr</code>	\wr	<code>\sqcap</code>	\sqcap
<code>\circ</code>	\circ	<code>\bigcirc</code>	\bigcirc	<code>\sqcup</code>	\sqcup
<code>\diamond</code>	\diamond	<code>\wedge</code>	\wedge	<code>\triangleleft</code>	\triangleleft
<code>\bullet</code>	\bullet	<code>\vee</code>	\vee	<code>\triangleright</code>	\triangleright

Um den Laplace-Operator darzustellen, kann man statt `\bigtriangleup` auch `\Delta` verwenden oder man benutzt `\nabla^2` (∇^2).

Das im Menü **Einfügen** \triangleright **Sonderzeichen** verfügbare Zeichen **Menütrenner** ist das Zeichen `\triangleright`.

10.4 Selbst definierte Operatoren

Mit Hilfe des Befehls `\DeclareMathOperator` kann man im \LaTeX -Vorspann eigene Operatoren definieren. Das Befehlsschema dazu lautet:

`\DeclareMathOperator{neuer Befehl}{Anzeige}`

Anzeige können Zeichen und Symbole sein, die festlegen, wie der Operator im Ausdruck aussieht. Will man einen großen Operator definieren, setzt man hinter den

Befehl ein *. Zu allen selbst definierten großen Operatoren kann man Grenzen angeben, siehe Kap. 10.2.

Z. B. erzeugt die L^AT_EX-Vorspannzeile

`\DeclareMathOperator*{\Raute}{\blacklozenge}`

den Befehl `\Raute`, mit dem man einen großen Operator aufrufen kann, der aus dem Rautesymbol aus Kap. 13.2 besteht:

$$\prod_{n=1}^{\infty}$$

Der Befehl für diese Formel lautet: `\Raute^{\infty}_{n=1}`

Benötigt man eigene Operatoren nicht mehrfach im Dokument, kann man diese auch mit den Befehlen `\mathop` und `\mathbin` definieren, die folgendes Schema besitzen:

`\mathop{Zeichen}` bzw. `\mathbin{Zeichen}`

Es können ein oder mehrere beliebige Zeichen angegeben werden. `\mathop` definiert große Operatoren, `\mathbin` binäre Operatoren.

`\mathop` lässt sich z. B. nutzen, um eine Grenze für mehrere Operatoren zu verwenden:

$$\sum_{i,j=1}^N$$

Der Befehl für obige Formel lautet: `\mathop{\sum\negmedspace\sum}_{i,j=1}^N`

11 Schriften

11.1 Schriftstile

Lateinische Buchstaben in einer Formel können in folgenden Schriftstilen gesetzt werden:

Befehl	Ergebnis	Tastenkürzel
<code>\mathbb{ABC}</code>	A B C	Alt-z k
<code>\mathbf{AbC}</code>	A b C	Alt-z f
<code>\boldsymbol{AbC}</code>	A b C	-
<code>\mathcal{ABC}</code>	<i>A</i> <i>B</i> <i>C</i>	Alt-z h
<code>\mathfrak{AbC}</code>	<i>A</i> <i>b</i> <i>C</i>	-

Befehl	Ergebnis	Tastenkürzel
<code>\mathit{AbC}</code>	<i>A</i> <i>b</i> <i>C</i>	-
<code>\mathrm{AbC}</code>	A b C	Alt-z r
<code>\mathsf{AbC}</code>	A b C	Alt-z s
<code>\mathtt{AbC}</code>	A b C	-

Dabei ist zu beachten, dass die Stile `\mathbb` und `\mathcal` nur für Großbuchstaben verwendet werden können.

Voreingestellt ist der Stil `\mathnormal`.

Die Stilbefehle wirken auch auf Buchstaben in mathematischen Konstruktionen:

$$\mathfrak{a} = \frac{\mathfrak{b}}{\mathfrak{c}}$$

Zeichen im mathematischen Textmodus erscheinen nicht in einem der Matheschriftstile, sondern in dem Textschriftstil `\textrm`.

Anstelle der Befehle kann man auch das Fenster **Mathe-Kontrollfläche** \triangleright **Mathe Stile & Schriften** verwenden.

11.2 Fett gedruckte Formeln

Möchte man komplette Formeln fett setzen, kann man nicht den im vorigen Kapitel angegebenen Befehl `\mathbf` verwenden. Denn dieser wirkt nicht auf kleine griechische Buchstaben. Außerdem setzt er lateinische Buchstaben immer aufrecht, was man an folgender Formel gut erkennt:

$$\int_n^2 \mathbf{f}(\theta) = \mathbf{\Gamma} \quad \text{Formel mit \code{\mathbf}}$$

Um die Formel korrekt darzustellen, verwendet man den Befehl `\boldsymbol`.

$$\int_n^2 \boldsymbol{f}(\boldsymbol{\theta}) = \boldsymbol{\Gamma} \quad \text{Formel mit \code{\boldsymbol}}$$

Es gibt auch die Möglichkeit die Formel in eine **boldmath-Umgebung** zu setzen. Diese Umgebung wird erzeugt, indem man im \TeX -Modus den Befehl `\boldmath` eingibt. Um die Umgebung wieder zu verlassen, gibt man im \TeX -Modus den Befehl `\unboldmath` ein.

$$\int_n^2 \boldsymbol{f}(\boldsymbol{\theta}) = \boldsymbol{\Gamma} \quad \text{Formel in boldmath-Umgebung}$$

11.3 Schriftgrößen

Für Zeichen in einer Formel gibt es analog zu Textzeichen folgende Größenbefehle:

`\Huge`, `\huge`, `\LARGE`, `\Large`, `\large`, `\normalsize`, `\small`, `\footnotesize`, `\scriptsize` und `\tiny`

Welche Zeichengröße die Befehle produzieren, hängt von der Dokumentschriftgröße ab, die dem Befehl `\normalsize` entspricht. Die anderen Befehle produzieren kleinere oder größere Schriften als `\normalsize`. Allerdings kann die Schriftgröße einen bestimmten Wert nicht übertreffen. Beträgt z. B. die Dokumentschriftgröße 12 pt, produziert der Befehl `\Huge` dasselbe wie `\huge`.

Ein Größenbefehl wird im T_EX-Modus vor der Formel eingegeben und legt die Größe aller nachfolgenden Formel- und Textzeichen fest. Um wieder zur Ausgangsgröße zurückzukehren, gibt man im T_EX-Modus `\normalsize` an.

Innerhalb einer Formel kann die Größe nur für Symbole oder Buchstaben im mathematischen Text geändert werden. Dazu wird der Größenbefehl im mathematischen Textmodus eingegeben. Alle diesem Befehl folgenden Zeichen bis zum Ende des mathematischen Textes oder bis zu einem anderen Größenbefehl haben dann die angegebene Größe. Dazu zwei Beispiele:

$$A = \frac{B}{c} \cdot \text{✠}$$

$$\text{✠}A\text{✠}A\text{✠}_{A\text{✠}}$$

Vor beiden Formeln wurde der Befehl `\huge` eingegeben. Der Befehl für die zweite Formel lautet: `\maltese_A Alt-m m \Large_\maltese_\rightarrow A \maltese_A Alt-m m \tiny_A \maltese`

Kann ein Symbol nicht in verschiedenen Größen dargestellt werden, wird es immer in der Standardgröße dargestellt.

12 Griechische Buchstaben

12.1 Kleine Buchstaben

Befehl	Ergebnis	Befehl	Ergebnis	Befehl	Ergebnis
<code>\alpha</code>	α	<code>\iota</code>	ι	<code>\varrho</code>	ϱ
<code>\beta</code>	β	<code>\kappa</code>	κ	<code>\sigma</code>	σ
<code>\gamma</code>	γ	<code>\varkappa</code>	\varkappa	<code>\varsigma</code>	ς
<code>\delta</code>	δ	<code>\lambda</code>	λ	<code>\tau</code>	τ
<code>\epsilon</code>	ϵ	<code>\mu</code>	μ	<code>\upsilon</code>	υ
<code>\varepsilon</code>	ε	<code>\nu</code>	ν	<code>\phi</code>	ϕ
<code>\zeta</code>	ζ	<code>\xi</code>	ξ	<code>\varphi</code>	φ
<code>\eta</code>	η	<code>o</code>	o	<code>\chi</code>	χ
<code>\theta</code>	θ	<code>\pi</code>	π	<code>\psi</code>	ψ
<code>\vartheta</code>	ϑ	<code>\varpi</code>	ϖ	<code>\omega</code>	ω
		<code>\rho</code>	ρ		

Eine Besonderheit stellt das μ dar, denn man kann es mit **AltGr-m** auch in einen Text eingeben. Dort erscheint es allerdings kursiv. Wie man ein aufrechtes μ im Text erzeugt, ist in Kap. 22.12 erklärt. Wie man generell aufrechte griechische Buchstaben erstellt, ist in Kap. 22.10 beschrieben.

12.2 Große Buchstaben

Befehl	Ergebnis	Befehl	Ergebnis
<code>\Gamma</code>	Γ	<code>\Sigma</code>	Σ
<code>\Delta</code>	Δ	<code>\Upsilon</code>	Υ
<code>\Theta</code>	Θ	<code>\Phi</code>	Φ
<code>\Lambda</code>	Λ	<code>\Psi</code>	Ψ
<code>\Xi</code>	Ξ	<code>\Omega</code>	Ω
<code>\Pi</code>	Π		

Um kursive Großbuchstaben zu erhalten, beginnt man einen Befehl mit **var**. So ergibt z. B. der Befehl `\varGamma`: Γ

12.3 Fett gedruckte Buchstaben

Griechische lassen sich nicht wie lateinische Buchstaben in verschiedenen Schriftstilen darstellen. Es besteht nur die Möglichkeit, sie mit dem Befehl `\boldsymbol` fett gedruckt auszugeben.

Befehl	Ergebnis
<code>\Upsilon\boldsymbol{\Upsilon}</code>	$\Upsilon\Upsilon$
<code>\theta\boldsymbol{\theta}</code>	$\theta\theta$

13 Symbole²³

13.1 Mathematische Symbole

Befehl	Ergebnis	Befehl	Ergebnis	Befehl	Ergebnis
<code>\neg</code>	\neg	<code>\forall</code>	\forall	<code>\prime</code>	$'$
<code>\Im</code>	\Im	<code>\exists</code>	\exists	<code>\backprime</code>	\backprime
<code>\Re</code>	\Re	<code>\nexists</code>	\nexists	<code>\mho</code>	\mho
<code>\aleph</code>	\aleph	<code>\emptyset</code>	\emptyset	<code>\triangle</code>	\triangle
<code>\partial</code>	∂	<code>\varnothing</code>	\varnothing	<code>\angle</code>	\angle
<code>\infty</code>	∞	<code>\dag</code>	\dagger	<code>\measuredangle</code>	\measuredangle
<code>\wp</code>	\wp	<code>\ddag</code>	\ddagger	<code>\sphericalangle</code>	\sphericalangle
<code>\imath</code>	\imath	<code>\complement</code>	\complement	<code>\top</code>	\top
<code>\jmath</code>	\jmath	<code>\Bbbk</code>	\mathbb{k}	<code>\bot</code>	\bot

13.2 Sonstige Symbole

Befehl	Ergebnis	Befehl	Ergebnis	Befehl	Ergebnis
<code>\flat</code>	\flat	<code>\hbar</code>	\hbar	<code>\diamondsuit</code>	\diamondsuit
<code>\natural</code>	\natural	<code>\hslash</code>	\hslash	<code>\Diamond</code>	\Diamond
<code>\sharp</code>	\sharp	<code>\clubsuit</code>	\clubsuit	<code>\heartsuit</code>	\heartsuit
<code>\surd</code>	\surd	<code>\spadesuit</code>	\spadesuit	<code>\P</code>	\P
<code>\checkmark</code>	\checkmark	<code>\bigstar</code>	\bigstar	<code>\copyright</code>	\copyright
<code>\yen</code>	\yen	<code>\blacklozenge</code>	\blacklozenge	<code>\circledR</code>	\circledR
<code>\pounds</code>	\pounds	<code>\blacktriangle</code>	\blacktriangle	<code>\maltese</code>	\maltese
<code>\\$</code>	$\$$	<code>\blacktriangledown</code>	\blacktriangledown	<code>\diagup</code>	\diagup
<code>\S</code>	\S	<code>\bullet</code>	\bullet	<code>\diagdown</code>	\diagdown

Weitere Symbole sind in Kap. 16.4 und 22.12 aufgelistet.

Einige Symbole können in unterschiedlicher Größe dargestellt werden, siehe Kap. 11.3.

13.3 Das EURO-Symbol

Wenn man in einer Formel das Eurosymbol verwenden will, muss man das \LaTeX -Paket **eurosym** installieren und dieses mit der \LaTeX -Vorspannzeile

²³Eine Liste mit sämtlichen Symbolen der meisten \LaTeX -Pakete findet man in [5].

`\usepackage[gennarrow]{eurosym}`

aktivieren. Das Eurosymbol kann man nun mit dem Befehl `\euro` aufrufen.

Das Eurozeichen im Text erhält man, wenn man `\euro` im T_EX-Modus eingibt²⁴. Das offizielle Währungssymbol erhält man über den Befehl `\officialeguro`, der nur im T_EX-Modus verfügbar ist.

Eine Übersicht der verschiedenen Eurosymbole:

	Befehl	Ergebnis
Formelmodus	<code>\euro</code>	€
T _E X-Modus	<code>\euro</code>	€
	<code>\officialeguro</code>	€

14 Relationen

Befehl	Ergebnis	Befehl	Ergebnis	Befehl	Ergebnis
<code><</code>	$<$	<code>=</code>	$=$	<code>></code>	$>$
<code>\le</code>	\leq	<code>\not=</code>	\neq	<code>\ge</code>	\geq
<code>\ll</code>	\ll	<code>\equiv</code>	\equiv	<code>\gg</code>	\gg
<code>\prec</code>	\prec	<code>\sim</code>	\sim	<code>\succ</code>	\succ
<code>\preceq</code>	\preceq	<code>\simeq</code>	\simeq	<code>\succeq</code>	\succeq
<code>\subset</code>	\subset	<code>\approx</code>	\approx	<code>\supset</code>	\supset
<code>\subseteq</code>	\subseteq	<code>\cong</code>	\cong	<code>\supseteq</code>	\supseteq
<code>\sqsubseteq</code>	\sqsubseteq	<code>\bowtie</code>	\bowtie	<code>\sqsupseteq</code>	\sqsupseteq
<code>\in</code>	\in	<code>\notin</code>	\notin	<code>\ni</code>	\ni
<code>\vdash</code>	\vdash	<code>\perp</code>	\perp	<code>\dashv</code>	\dashv
<code>\smile</code>	\smile	<code>\propto</code>	\propto	<code>\frown</code>	\frown
<code>\lhd</code>	\lhd	<code>\asymp</code>	\asymp	<code>\rhd</code>	\rhd
<code>\unlhd</code>	\unlhd	<code>\doteq</code>	\doteq	<code>\unrhd</code>	\unrhd
<code>\gtrless</code>	\gtrless	<code>\circeq</code>	\circeq	<code>\lessgtr</code>	\lessgtr
<code>\mid</code>	\mid	<code>\models</code>	\models	<code>\parallel</code>	\parallel
<code>\nmid</code>	\nmid	<code>\widehat{=}</code>	$\widehat{=}$	<code>\nparallel</code>	\nparallel

Die Zeichen `\lhd` bzw. `\rhd` sind größer als die gleich aussehenden Operatoren `\triangleleft` bzw. `\triangleright`.

Relationen sind im Gegensatz zu Symbolen immer von Leerraum umgeben.

²⁴Ein L_YX Fehler verhindert, dass das €-Zeichen direkt über die Tastatur eingegeben werden kann, siehe [L_YX-Fehlernummer #845](#).

Relationen mit Beschriftungen können mit Hilfe des Befehls `\stackrel` erstellt werden:

Befehl	Ergebnis
<code>A(r)\stackrel{r}{\approx} B</code>	$A(r) \stackrel{r}{\approx} B$

15 Funktionen

15.1 Vordefinierte Funktionen

Im Allgemeinen werden in mathematischen Ausdrücken Buchstaben *kursiv* gesetzt, Funktionsnamen hingegen nicht. Aus diesem Grund gibt es vordefinierte Funktionen, die zusätzlich von Vorfaktoren etwas abgesetzt sind. Um diese aufzurufen, wird vor den Funktionsnamen ein Backslash gesetzt.

Befehl	Ergebnis	Befehl	Ergebnis
<code>A\sin(x)+B</code>	$A\sin(x) + B$	<code>A\sin(x)+B</code>	$A\sin(x) + B$

Folgende Funktionen sind vordefiniert:

Befehl	Befehl	Befehl	Befehl
<code>\sin</code>	<code>\sinh</code>	<code>\arcsin</code>	<code>\sup</code>
<code>\cos</code>	<code>\cosh</code>	<code>\arccos</code>	<code>\inf</code>
<code>\tan</code>	<code>\tanh</code>	<code>\arctan</code>	<code>\lim</code>
<code>\cot</code>	<code>\coth</code>	<code>\arg</code>	<code>\liminf</code>
<code>\sec</code>	<code>\min</code>	<code>\deg</code>	<code>\limsup</code>
<code>\csc</code>	<code>\max</code>	<code>\det</code>	<code>\Pr</code>
<code>\ln</code>	<code>\exp</code>	<code>\dim</code>	<code>\hom</code>
<code>\lg</code>	<code>\log</code>	<code>\ker</code>	<code>\gcd</code>

15.2 Selbst definierte Funktionen

Will man eine Funktion verwenden, die nicht vordefiniert ist, wie z. B. die Signumfunktion $\operatorname{sgn}(x)$, hat man zwei Möglichkeiten:

- Man definiert die Funktion selbst. Dazu gibt man im \LaTeX -Vorspann die Zeile
`\DeclareMathOperator{\sgn}{sgn}`

ein²⁵. Nun kann man mit dem Befehl `\sgn` die neu definierte Funktion aufrufen.

²⁵Näheres zu `\DeclareMathOperator` siehe Kap. 10.4.

- Man schreibt die Formel wie gewohnt, markiert den Formelnamen, also in unserem Beispiel die Buchstaben *sgn*, und wandelt ihn in in mathematischen Text um. Zwischen Vorfaktor und Funktion fügt man ein geschütztes Leerzeichen ein.

Das Ergebnis ist bei beiden Methoden dasselbe wie bei einer vordefinierten Funktion²⁶:

Befehl	Ergebnis
$A \backslash \text{sgn}(x) + B$	$A \operatorname{sgn}(x) + B$
$A \backslash, \underbrace{\text{sgn}}_{\text{Alt-m m}}(x) + B$	$A \operatorname{sgn}(x) + B$

15.3 Grenzwerte

Für Grenzwerte sind außer $\backslash \mathbf{lim}$, $\backslash \mathbf{liminf}$ und $\backslash \mathbf{limsup}$ noch folgende Funktionen definiert:

Befehl	Ergebnis
$\backslash \text{varliminf}$	$\underline{\underline{\lim}}$
$\backslash \text{varlimsup}$	$\overline{\overline{\lim}}$
$\backslash \text{varprojlim}$	\varprojlim
$\backslash \text{varinjlim}$	\varinjlim

Der Grenzwert wird durch Tiefstellen erzeugt. Bei einer eingebetteten Formel wird er rechts neben die Funktion gesetzt:

Befehl	Ergebnis
$\backslash \lim_{x \rightarrow A} x = B$	$\lim_{x \rightarrow A} x = B$

In einer abgesetzten Formel steht der Grenzwert wie gewohnt darunter:

$$\lim_{x \rightarrow A} x = B$$

²⁶Selbst definierte Funktionen werden in L^AT_EX rot, vordefinierte schwarz angezeigt.

15.4 Modulo-Funktionen

Eine Besonderheit stellt die Modulo-Funktion dar, denn es gibt sie in vier Varianten. Die Funktionsvarianten in einer abgesetzten Formel:

Befehl	Ergebnis
$a \backslash \text{mod}_b$	$a \mod b$
$a \backslash \text{pmod}_b$	$a \pmod{b}$
$a \backslash \text{bmod}_b$	$a \bmod b$
$a \backslash \text{pod}_b$	$a \pmod{b}$

In einer eingebetteten Formel wird bei allen Varianten vor den Funktionsnamen weniger Leerraum gesetzt.

16 Sonderzeichen

16.1 Sonderzeichen im mathematischen Text

Die folgenden Befehle können nur im mathematischen Textmodus oder im $\text{T}_{\text{E}}\text{X}$ -Modus verwendet werden:

Befehl	Ergebnis
$\backslash \text{oe}$	œ
$\backslash \text{OE}$	Œ
$\backslash \text{ae}$	æ
$\backslash \text{AE}$	Æ
$\backslash \text{aa}$	å
$\backslash \text{AA}$	Å
$\backslash \text{i}$	ı

Befehl	Ergebnis
$\backslash \text{o}$	ø
$\backslash \text{O}$	Ø
$\backslash \text{l}$	ł
$\backslash \text{L}$	Ł
$\text{!}'_{\text{L}}$	ı
'?_{L}	ı
$\backslash \text{j}$	Ј

Die Zeichen Å und Ø können auch über das Fenster **Mathe-Kontrollfläche** eingefügt werden.

Eine Ausnahme bilden die Befehle $\text{!}'_{\text{L}}$ und '?_{L} , denn man kann sie in $\text{L}_{\text{X}}\text{X}$ direkt in einen Text eingeben.

16.2 Akzente im Text

Man kann mit den folgenden Befehlen sämtliche Buchstaben akzentuieren. Die Befehle müssen dazu im $\text{T}_{\text{E}}\text{X}$ -Modus eingegeben werden.

Befehl	Ergebnis
<code>\“e</code>	ë
<code>\‘e</code>	è
<code>\^_e</code>	ê
<code>\=e</code>	ē
<code>\u_e</code>	ě
<code>\b_e</code>	ē
<code>\t_ee</code>	ēē

Befehl	Ergebnis
<code>\H_e</code>	é
<code>\’e</code>	é
<code>\~e</code>	ẽ
<code>\.e</code>	è
<code>\v_e</code>	ě
<code>\d_e</code>	ē
<code>\c_e</code>	ç

Mit dem Befehl `\t` können auch zwei verschiedene Zeichen akzentuiert werden. Der Befehl `\t_sz` ergibt z. B. $\hat{s}z$.

Die Akzente ‘, ’ und ^ können in Verbindung mit Selbstlauten auch direkt über die Tastatur eingegeben werden ohne den T_EX-Modus zu benutzen. Dasselbe gilt für die Tilde²⁷ ~ in Verbindung mit *a*, *n* oder *o*.

Die Befehle `\b`, `\c`, `\d`, `\H`, `\t`, `\u`, `\v` und direkt über die Tastatur eingegebene Akzente sind auch im mathematischen Textmodus verfügbar. Für die restlichen Akzente gibt es in Formeln spezielle Mathebefehle, siehe Kap. 7.1.

Des Weiteren kann man mit dem Befehl `\textcircled`, ähnlich dem Copyrightsymbol, sämtliche Zahlen und Buchstaben in einen Kreis setzen, quasi mit einem Kreis akzentuieren.

Befehl	Ergebnis
<code>\textcircled{w}</code>	Ⓜ
<code>\Large \textcircled{\normalsize\raisebox{-1.5pt}{W}}</code>	Ⓜ

Man muss nur darauf achten, dass das Zeichen in einen Kreis passt. `\Large`²⁸ bestimmt dabei die Größe des Kreises. Mit Hilfe von `\raisebox`²⁹ kann man das Zeichen zentrieren.

16.3 Minuskelziffern

Minuskelziffern werden mit dem Befehl `\oldstylenums` erzeugt. Der Befehl ist sowohl in einer Formel als auch im T_EX-Modus verwendbar. Das Befehlsschema lautet:

`\oldstylenums{Ziffer}`

Der Befehl `\oldstylenums{0123456789}` ergibt: 0123456789

²⁷Gilt nur für Tastaturen, bei denen die Tilde als Akzent definiert ist.

²⁸siehe Kap. 11.3

²⁹siehe Kap. 9.2

16.4 Sonstige Sonderzeichen

Folgende Zeichen können in eine Formel nur mit Befehlen eingegeben werden:

Befehl	Ergebnis
<code>\^_</code>	$\hat{_}$
<code>_</code>	$_$
<code>\^_ \circ</code>	$\hat{_}^\circ$

Bei der Verwendung von Apostrophen in Formeln kann folgendes Problem auftreten: Der Apostroph ist in \LaTeX als hochgestelltes Symbol `\prime`³⁰ definiert. Verwendet man Exponenten und Indizes zusammen mit Apostrophen, wie in „ $x_1'^2$ “, bedeutet der zugehörige \LaTeX -Befehl `x'_1^2` in \LaTeX `x^\prime_1^2` also einen doppelten Exponenten. Um den dadurch entstehenden \LaTeX -Fehler zu vermeiden, verwendet man statt dem Apostroph gleich ein hochgestelltes `\prime`.

Um Ableitungen mit Exponenten zu versehen, bietet sich an, die Ableitung in eine \LaTeX -Klammer zu setzen. So erzeugt z. B. der Befehl `\{x'_1 \rightarrow^2` Folgendes:² Da in diesem Fall der Apostroph durch eine \LaTeX -Klammer vom Exponenten getrennt ist, kann das oben beschriebene Problem nicht auftreten.

Die Tilde ist nur für Akzente gedacht, weswegen sie alleine deplatziert wirkt: \sim . In Webadressen (URLs) wird eine mittig platzierte Tilde benötigt. Man verwendet in diesem Fall das Relationszeichen `\sim`³¹ in einer Formel. Da in einer URL-Box (Menü **Einfügen** \triangleright **URL**) keine Formel erstellt werden kann, gibt man die URL als normalen Text in der Stilart 'Schreibmaschine' ein. Soll die URL im DVI- oder PDF-Dokument einen Link produzieren, verwendet man den Befehl `\href`³².

Als Beispiel eine URL mit einer Tilde:

als Text: `http://www.lyx.org/~mustermann`

mit `\href`: `\href{http://www.lyx.org/~mustermann}`

17 Formelstile

- Für abgesetzte Formeln gibt es zwei verschiedene Anordnungsstile:

Zentriert dies ist der voreingestellte Standard

³⁰`\prime` ist in Kap. 13.1 aufgelistet.

³¹siehe Kap. 14

³²`\href` kann nur zusammen mit dem \LaTeX -Paket **hyperref** verwendet werden. Mehr zu `\href` steht in Kap. 22.18.2.

Halblinks dazu muss im Menü **Dokument** ▸ **Einstellungen** unter **Dokument-**
klasse die Option **fleqn** eingegeben werden.

Wenn man **Halblinks** gewählt hat, kann man mit der Länge `\mathindent` den Abstand zwischen dem linken Seitenrand und den Formeln einstellen. Soll z. B. der Abstand 15 mm betragen, gibt man im L^AT_EX-Vorspann die Zeile

`\setlength{\mathindent}{15mm}`

ein. Wenn man keinen Abstand vorgibt, wird der voreingestellte Wert von 30 pt verwendet.

- Und zwei verschiedene Nummerierungsstile:

Rechts dies ist der voreingestellte Standard

Links dazu muss im Menü **Dokument** ▸ **Einstellungen** unter **Dokument-**
klasse die Option **leqno** eingegeben werden.

fleqn und **leqno** können auch gemeinsam verwendet werden. Dazu werden beide Befehle, durch ein Komma getrennt, hintereinander geschrieben.

Die eingestellten Stile gelten immer für alle abgesetzten Formeln eines Dokuments. Möchte man jedoch in einem Dokument sowohl zentrierte als auch halblinks ausgerichtete Formeln setzen, verwendet man den Stil **Zentriert**. Die halblinks auszurichtenden Formeln setzt man dann in eine flalign-Umgebung, siehe Kap. 18.2.3.

18 Mehrzeilige Formeln

18.1 Allgemeines

In L^yX werden mehrzeilige Formeln erstellt, wenn man in einer Formel **Strg-Enter** drückt. Man befindet sich nun in einer **eqnarray-Umgebung**.

Es gibt noch weitere mehrzeilige Formelumgebungen, die man über das Menü **Einfügen** ▸ **Mathe** erstellt. Diese Umgebungen werden in den weiteren Kapiteln beschrieben.

In allen mehrzeiligen Formelumgebungen erstellt man eine neue Zeile, indem man ein weiteres Mal **Strg-Enter** drückt.

18.1.1 Eqnarray-Umgebung

Drückt man in einer leeren Formel **Strg-Enter**, erscheinen drei blaue Kästchen.³³ Der Inhalt des ersten Kästchens wird rechtsbündig, der des letzten linksbündig ausgerichtet. Der Inhalt des mittleren Kästchens erscheint zentriert und etwas kleiner,

³³Durch einen Fehler kann es sein, dass statt einer eqnarray-Umgebung eine align-Umgebung erstellt wird, siehe [L^yX-Fehlernummer #2543](#).

da in dieses nur die Relation, wie z. B. das \leq Zeichen, eingegeben wird.

$$\frac{ABC}{D} \quad \frac{ABC}{D} \quad \frac{ABC}{D}$$

$$AB \quad AB \quad AB$$

$$A \quad = \quad A$$

In der eqnarray-Umgebung ist es nicht möglich mehrere Formeln nebeneinander zu setzen. Man verwendet in diesem Fall eine Align-Umgebung, siehe Kap. 18.2.

18.1.2 Zeilenabstand

Bei mehrzeiligen Formeln fehlt manchmal etwas Platz zwischen den Zeilen:

$$B^2(B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) + 4x_0^2x^2 + 4x_0xD = -4x^2B^2 + 4x_0xB^2$$

$$4x^2(B^2 + x_0^2) + 4x_0x(D - B^2) + B^2(B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) = 0$$

In L^AT_EX wird zusätzlicher Leerraum in Formeln als optionales Argument des Zeilennumbruchbefehls angegeben. Dies ist in L_YX aber nicht möglich³⁴, weshalb man die ganze Formel im T_EX-Modus eingeben muss. Um für unser Beispiel Leerraum einzufügen, gibt man am Ende der ersten Zeile den Befehl `\[3mm]` ein. Damit erhält man:

$$B^2(B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) + 4x_0^2x^2 + 4x_0xD = -4x^2B^2 + 4x_0xB^2$$

$$4x^2(B^2 + x_0^2) + 4x_0x(D - B^2) + B^2(B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) = 0$$

Möchte man den Zeilenabstand für alle Zeilen einer Formel festlegen, verändert man die Länge `\jot`. Es gilt `Zeilenabstand = 6 pt + \jot`. Voreingestellt ist für `\jot` der Wert 3 pt. Um wie im vorigen Beispiel 3 mm zusätzlichen Zeilenabstand zu erzeugen, gibt man den Befehl

`\setlength{\jot}{3mm+3pt}`

im T_EX-Modus vor der Formel ein. Das setzt voraus, dass das Paket **calc**³⁵ im L^AT_EX-Vorspann mit der Zeile

`\usepackage{calc}`

geladen wurde.

³⁴ siehe [L_YX-Fehlernummer #1505](#)

³⁵ **calc** ist Teil jeder L^AT_EX-Standardinstallation.

Man erhält:

$$\begin{aligned} B^2(B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) + 4x_0^2x^2 + 4x_0xD &= -4x^2B^2 + 4x_0xB^2 \\ 4x^2(B^2 + x_0^2) + 4x_0x(D - B^2) + B^2(B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) &= 0 \end{aligned}$$

Um wieder zum voreingestellten Abstand zu wechseln, setzt man `\jot` auf den Wert 3pt.

18.1.3 Spaltenabstand

Mehrzeilige Formeln bilden eine Matrix. Eine Formel in der Eqnarray-Umgebung ist z. B. eine dreispaltige Matrix. Durch Änderung des Spaltenabstands in dieser Umgebung, kann man den Leerraum neben den Relationszeichen ändern.

Der Spaltenabstand wird über die Länge `\arraycolsep` festgelegt, wobei gilt: Spaltenabstand = 2 `\arraycolsep`. Der im TeX-Modus eingegebene Befehl

`\setlength{\arraycolsep}{1cm}`

bewirkt also für alle folgenden Formeln einen Spaltenabstand von 2 cm. Um wieder zum voreingestellten Abstand zurückzukehren, setzt man `\arraycolsep` auf 5 pt.

Eine Formel mit 2 cm Spaltenabstand:

$$\begin{array}{ccc} A & = & B \\ C & \neq & A \end{array}$$

Eine Formel mit dem für Matrizen voreingestellten Spaltenabstand von 10 pt:

$$\begin{array}{ccc} A & = & B \\ C & \neq & A \end{array}$$

18.1.4 Lange Formeln

Lange Formeln kann man wie folgt setzen:

- Ist eine Seite der Gleichung deutlich kürzer als die Zeilenbreite, wählt man man diese als linke Seite und schreibt die rechte über zwei Zeilen:

$$\begin{aligned} H &= W_{SB} + W_{mv} + W_D - \frac{\hbar^2}{2m_0}\Delta - \frac{\hbar^2}{2m_1}\Delta_1 - \frac{\hbar^2}{2m_2}\Delta_2 - \frac{e^2}{4\pi\epsilon_0|\mathbf{r} - \mathbf{R}_1|} \\ &\quad - \frac{e^2}{4\pi\epsilon_0|\mathbf{r} - \mathbf{R}_2|} + \frac{e^2}{4\pi\epsilon_0|\mathbf{R}_1 - \mathbf{R}_2|} \end{aligned} \quad (7)$$

Das Minuszeichen zu Beginn der zweiten Zeile erscheint eigentlich nicht als Operator, da es das erste Zeichen der Zeile ist. Deswegen wäre es nicht von Leerraum umgeben und ließe sich schlecht vom Bruchstrich unterscheiden. Um das zu verhindern, wurde mit dem Befehl `\hspace{3pt}`³⁶ Leerraum hinter das Minuszeichen eingefügt.

- Sind beide Seiten der Gleichung zu groß, verwendet man den Befehl `\lefteqn`. Dieser wird in das erste blaue Kästchen der ersten Zeile eingegeben und bewirkt, dass alle weiteren Zeilen im Vergleich zur ersten eingerückt sind:

$$\begin{aligned}
 & 4x^2 (B^2 + x_0^2) + 4x_0x (D - B^2) + B^2 (B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) + D^2 \\
 & - B^2 - 2B\sqrt{r_g^2 - x^2 + 2x_0x - x_0^2 + r_g^2 - x^2 + 2x_0x - x_0^2} \\
 & = B^2 + 2(r_g^2 + 2x_0x - x_0^2 - r_k^2) + \frac{(r_g^2 + 2x_0x - x_0^2 - r_k^2)^2}{B^2} \quad (8)
 \end{aligned}$$

Hat man `\lefteqn` eingegeben, befindet sich der Cursor in einem pinkfarbenen Kästchen, das gegenüber dem blauen leicht verschoben ist. In dieses wird die weitere Formel eingegeben. Dabei werden die anderen blauen Kästchen überschrieben. Das ist gewollt, da dies auch im Ausdruck geschieht.

Der Inhalt der weiteren Zeilen wird entweder in das zweite oder dritte Kästchen eingegeben. Bei Eingabe in das dritte Kästchen ist die Einrückung etwas größer.

Bei der Verwendung von `\lefteqn` ist Folgendes zu beachten:

- * Die Formel nutzt nicht die gesamte Seitenbreite aus. Würde man in obigem Beispiel der ersten Zeile den Term $-B^2$ anfügen, befände sich dieser außerhalb des Seitenrands. Um die Breite besser auszunutzen, kann man zu Beginn der ersten Zeile negativen Leerraum einfügen.
 - * Ragt das Ende der ersten Zeile in LyX über das der anderen Zeilen hinaus, kann der Cursor nicht mit der Maus in den hinausragenden Teil gesetzt werden. Man klickt in diesem Fall in die Mitte der ersten Zeile und bewegt den Cursor mit den Pfeiltasten³⁷.
- Weitere Möglichkeiten lange Formeln zu setzen, bieten die in Kap. 18.4 und 18.5 beschriebenen Umgebungen.

³⁶mehr zu `\hspace` siehe Kap. 8.2

³⁷[LyX-Fehlernummer #1429](#)

18.1.5 Mehrzeilige Klammern

Bei Klammern, die über mehrere Zeilen gehen, tritt folgendes Problem auf:

$$A = \sin(x) \left[\prod_{R=1}^{\infty} \frac{1}{R} + \cdots \right. \\ \left. \cdots + B - D \right]$$

Klammern variabler Größe dürfen nicht über mehrere Zeilen gehen. Deswegen beendet man die erste Zeile mit `\right.` und beginnt die zweite Zeile mit `\left.`³⁸.

Die Klammer der zweiten Zeile sollte genauso groß wie die der ersten sein. Um das zu erreichen, gibt man nach `\left.` den Befehl `\vphantom{\prod^{\infty}_{R=1}}` ein. Denn das Multiplikationszeichen mit seinen Grenzen ist das größte Symbol der ersten Zeile, und daran soll die zweite Klammer angepasst werden.

Das Ergebnis sieht so aus:

$$A = \sin(x) \left[\prod_{R=1}^{\infty} \frac{1}{R} + \cdots \right. \\ \left. \cdots + B - D \right]$$

18.2 Align-Umgebungen

Um mehrere Formeln nebeneinander setzen zu können, verwendet man die Align-Umgebungen.

Align-Umgebungen bestehen aus Spalten. Die ungeradzahligen Spalten sind rechts-, die geradzahligen linksbündig ausgerichtet. Es gibt also keine spezielle Spalte für das Relationszeichen. Des Weiteren gibt es im Gegensatz zur eqnarray-Umgebung keinen voreingestellten Spaltenabstand.

Jede Zeile einer Align-Umgebung kann nummeriert werden.

Align-Umgebungen werden über das Menü **Einfügen** \triangleright **Mathe** erstellt. Über das Menü **Bearbeiten** \triangleright **Mathe** \triangleright **Formelart ändern** können bereits existierende Formeln in Align-Umgebungen umgewandelt werden.

Um Spalten hinzuzufügen oder zu entfernen, verwendet man die Tabellen-Symbolleiste, das Menü **Bearbeiten** \triangleright **Zeilen & Spalten** oder die Tastenkürzel **Alt-m c i** bzw. **Alt-m c d**.

³⁸mehr zu `\left` und `\right` siehe Kap. 5.1.2

18.2.1 Standard align-Umgebung

Diese Align-Umgebung wird über das Menü **Einfügen** ▸ **Mathe** ▸ **AMS align-Umgebung** erstellt.

Ein Beispiel für zwei nebeneinander stehende Formeln, die mit einer vierspaltigen align-Umgebung erzeugt wurden:

$$\begin{array}{cc} A = \sin(B) & C = D \\ C \neq A & B \neq D \end{array}$$

Wie man sieht, werden die Formeln in dieser Umgebung so angeordnet, als würde sich vor der ersten und hinter jeder geraden Spalte ein `\hfill`³⁹ befinden. Ist der Formelstil **Halblinks**⁴⁰ gewählt, wird die Formel ohne den `\hfill` vor der ersten Spalte gesetzt.

18.2.2 Alignat-Umgebung

In der alignat-Umgebung gibt es keinen Spaltenabstand. Man fügt ihn manuell durch die in Kap. 8 beschriebenen Leerräume ein.

Das vorige Beispiel in der alignat-Umgebung, bei dem zu Beginn der zweiten Formel 1 cm Leerraum eingefügt wurde:

$$\begin{array}{cc} A = \sin(B) & C = D \\ C \neq A & B \neq D \end{array}$$

Weil man den Spaltenabstand für jede Spalte einzeln einstellen kann, eignet sich diese Umgebung besonders für drei und mehr nebeneinander gesetzte Formeln.

18.2.3 Flalign-Umgebung

Bei dieser Umgebung werden die ersten beiden Spalten immer so weit links und die letzte Spalte so weit rechts wie möglich gesetzt. Dazu ein Beispiel:

$$\begin{array}{ccc} A = 1 & B = 2 & C = 3 \\ X = -1 & Y = -2 & Z = 4 \end{array}$$

Erzeugt man eine flalign-Umgebung mit ungerader Spaltenanzahl und gibt in die letzte Spalte eine leere \LaTeX -Klammer ein, kann man einzelne Formeln im Dokument

³⁹mehr zu `\hfill` siehe Kap. 8.2.

⁴⁰Formelstile siehe Kap. 17

links setzen obwohl der Formelstil **Zentriert** eingestellt ist. Als Beispiel die halblinks ausgerichtete Formel (5):

$$\iiint_V X \, dV = U \quad (9)$$

In den ersten beiden Spalten befindet sich die Formel. Damit sie wie beim Formelstil **Halblinks** vom Rand etwas abgesetzt ist, wurde zu Beginn der ersten Spalte 30 pt Leerraum eingefügt.

18.3 Gather-Umgebung

Diese Umgebung besteht nur aus einer Spalte, deren Inhalt zentriert ist. Jede Zeile kann nummeriert werden.

$$A = 1 \quad (10)$$

$$X = -1 \quad (11)$$

18.4 Multline-Umgebung

Die multiline-Umgebung besteht wie die Gather-Umgebung nur aus einer Spalte. Jedoch ist die erste Zeile links, die letzte rechts ausgerichtet. Alle anderen Zeilen sind zentriert. Dadurch eignet sich diese Umgebung besonders für lange Formeln. Als Beispiel Formel (8) in der multiline-Umgebung:

$$\begin{aligned} &4x^2 (B^2 + x_0^2) + 4x_0x (D - B^2) + B^2 (B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) + D^2 \\ &\quad - B^2 - 2B\sqrt{r_g^2 - x^2 + 2x_0x - x_0^2 + r_g^2 - x^2 + 2x_0x - x_0^2} \\ &\quad = B^2 + 2(r_g^2 + 2x_0x - x_0^2 - r_k^2) + \frac{(r_g^2 + 2x_0x - x_0^2 - r_k^2)^2}{B^2} \end{aligned} \quad (12)$$

In der Ausgabe erscheint nur die letzte (erste) Zeile einer multiline-Umgebung nummeriert, wenn rechts (links) nummeriert wird, auch wenn in L^AT_EX hinter jeder Zeile eine Formelnummer angezeigt wird.⁴¹

Mit den Befehlen `\shoveright` und `\shoveleft` kann eine zentrierte Zeile rechts bzw. links ausgerichtet werden. Die Befehle werden folgendermaßen verwendet:

`\shoveright{Zeileninhalt}` bzw. `\shoveleft{Zeileninhalt}`

Die Länge `\multlinegap` legt den Abstand der ersten Zeile vom linken Rand fest. Voreingestellt ist die Länge 0 pt.

⁴¹Numerierungsstile siehe Kap. Kap. 17

Als Beispiel obige Formel, vor der im T_EX-Modus der Befehl

`\setlength{\multlinegap}{2cm}`

einggegeben wurde:

$$\begin{aligned} & 4x^2 (B^2 + x_0^2) + 4x_0x (D - B^2) + B^2 (B^2 - 2r_g^2 + 2x_0^2 - 2r_k^2) + D^2 \\ & - B^2 - 2B\sqrt{r_g^2 - x^2 + 2x_0x - x_0^2 + r_g^2 - x^2 + 2x_0x - x_0^2} \\ & = B^2 + 2 (r_g^2 + 2x_0x - x_0^2 - r_k^2) + \frac{(r_g^2 + 2x_0x - x_0^2 - r_k^2)^2}{B^2} \end{aligned} \quad (13)$$

Die zweite Zeile wurde mit `\shoveleft` linksbündig ausgerichtet.

18.5 Mehrzeilige Formelteile

Möchte man nur Teile einer Formel mehrzeilig darstellen, verwendet man eine der folgenden Umgebungen: **aligned**, **alignedat**, **gathered** oder **split**. Sie können über das Menü **Einfügen** \triangleright **Mathe** oder, wie in diesem Abschnitt beschrieben, mit Befehlen erstellt werden.

Die ersten drei haben dieselben Eigenschaften wie die entsprechenden mehrzeiligen Formelumgebungen. Man kann jedoch zusätzlich weitere Formelteile daneben setzen. Dazu ein Beispiel:

$$\left. \begin{aligned} \Delta x \Delta p &\geq \frac{\hbar}{2} \\ \Delta E \Delta t &\geq \frac{\hbar}{2} \end{aligned} \right\} \text{Unbestimmtheitsrelationen}$$

Um diese Formel zu erhalten, wird eine abgesetzte Formel erstellt, in die der Befehl `\aligned` eingegeben wird. Es erscheint nun eine lila Box um das blaue Formelkästchen in der man Spalten und Zeilen hinzufügen kann. Des Weiteren kann man auch außerhalb der mehrzeiligen Umgebung Formelteile setzen, wie z. B. die Klammer.

Die `aligned`-Umgebung eignet sich auch für lange Formeln, deren Zeilen horizontal ausgerichtet sind. Verwendet man `aligned` in einer abgesetzten Formel, hat das den Vorteil, dass die Formelnummer mittig hinter den Zeilen steht. Als Beispiel Formel (7) in der `aligned`-Umgebung:

$$\begin{aligned} H = W_{SB} + W_{mv} + W_D - \frac{\hbar^2}{2m_0}\Delta - \frac{\hbar^2}{2m_1}\Delta_1 - \frac{\hbar^2}{2m_2}\Delta_2 - \frac{e^2}{4\pi\epsilon_0|\mathbf{r} - \mathbf{R}_1|} \\ - \frac{e^2}{4\pi\epsilon_0|\mathbf{r} - \mathbf{R}_2|} + \frac{e^2}{4\pi\epsilon_0|\mathbf{R}_1 - \mathbf{R}_2|} \end{aligned} \quad (14)$$

Möchte man die Umgebung **alignedat**, **gathered** oder **split** verwenden, gibt man den Befehl `\alignedat`, `\gathered` bzw. `\split` ein. Die `split`-Umgebung hat dieselben Eigenschaften wie die `aligned`-Umgebung, jedoch darf sie nur zwei Spalten enthalten.

18.6 Text in mehrzeiligen Formeln

Bei den `Align`-Umgebungen sowie der `multline`- und `gather`-Umgebung besteht die Möglichkeit Text einzufügen, der in einer eigenen Zeile steht und die Ausrichtung der Spalten nicht beeinflusst. Er wird erstellt, indem man am Zeilenende den Befehl `\intertext` eingibt. Dieser besitzt folgendes Schema:

`\intertext{Textinhalt}`

Der Textinhalt sollte nicht länger als eine Zeile sein, da er nicht umgebrochen werden kann. Weil `LyX` `\intertext` noch nicht unterstützt, schreibt man den Textinhalt als mathematischen Text. `\intertext` muss dabei zu Beginn einer Zeile stehen und erscheint in der Ausgabe über dieser Zeile. Ein Beispiel bei dem der Text zu Beginn der zweiten Zeile eingegeben wurde:

$$I = a\sqrt{2} \int_0^{2\pi} \sqrt{1 + \cos(\phi)} \, d\phi \quad (15)$$

Integrand ist symmetrisch zu $\phi = \pi$, daher

$$= 2a\sqrt{2} \int_0^{\pi} \sqrt{1 + \cos(\phi)} \, d\phi \quad (16)$$

19 Formelnummerierung

19.1 Allgemeines

Formeln werden über das Menü **Bearbeiten** \triangleright **Mathe** \triangleright **Nummerierung an/aus** (Tastenkürzel **Alt-m n**) nummeriert. Es erscheint dann in `LyX` hinter der Formel eine Raute in runden Klammern. Im Ausdruck erscheint anstelle der Raute die Formelnummer.

Wird in mehrzeiligen Formeln die Nummerierung angeschaltet, werden alle Zeilen nummeriert. Man kann jedoch über das Menü

Bearbeiten \triangleright **Mathe** \triangleright **Zeilennummerierung an/aus** (Tastenkürzel **Alt-m N**) für jede Zeile die Nummerierung festlegen.

Um automatisch nummerierte Formeln zu erstellen, kann man die folgende Zeile der Datei **cua.bind** hinzufügen, die sich in `LyX`s Installationsordner befindet, und anschließend `LyX` neu starten:

`\bind "C-h" "command-sequence math-display; math-number;"`

Eine automatisch nummerierte Formel wird dann durch das Tastenkürzel **Ctrl-h** erstellt.

Außer eingebetteten Formeln können alle Formeln in zwei verschiedenen Stilen nummeriert werden, siehe Kap. 17.

19.2 Querverweise

Es kann auf alle Formeln verwiesen werden, die eine Marke besitzen. Eine Marke wird über das Menü **Einfügen** \triangleright **Marke** gesetzt. Dabei muss sich der Cursor in einer abgesetzten Formel befinden. Es erscheint ein Eingabefeld mit dem voreingestellten Eintrag **eq:**, nachdem der eigentliche Markenname eingegeben wird. Der voreingestellte Eintrag steht für „equation“ und erleichtert bei größeren Dokumenten die Zuordnung, dass die Marke zu einer Formel und nicht zu einem Kapitel gehört. Um eine Marke zu ändern oder zu löschen, wählt man wieder das Menü **Einfügen** \triangleright **Marke**.

Der Name der Marke erscheint in LyX in zwei runden Klammern hinter der Formel. Eine Formel mit Marke wird immer nummeriert. Wird die Nummerierung abgeschaltet, wird die Marke gelöscht. Dies ist momentan die einzige Möglichkeit eine Marke zu löschen, siehe [LyX-Fehlernummer #2556](#).

Die Querverweise werden über das Menü **Einfügen** \triangleright **Querverweis** eingefügt. Ein Querverweis erscheint im Ausdruck als Formelnummer, wenn man jedoch im Querverweisdialogfenster das Format [**<Querverweis>**] wählt, erscheint der Querverweis als Formelnummer mit umgebenden Klammern. Der Name der grauen Querverweisbox lautet dann „Gleichung“ anstatt „Verweis“.

Drückt man in LyX mit der rechten Maustaste auf einen Verweis, gelangt man zur Formel auf die verwiesen wird.

Als Beispiel folgen Querverweise zu Formeln aus diesem Kapitel:

Die Gleichungen (etwas) und (17b) sind äquivalent. In (W) werden im Gegensatz zu (XXI) große lateinische Buchstaben zur Nummerierung verwendet.

Ist im Argument von `\tag`⁴² wie in Kap. 9.4 eine Box angegeben, kann nicht auf die Formel verwiesen werden.

19.3 Formeluntergliederung

Möchte man Formeln weiter untergliedern, kann man das mit Hilfe der Befehle `\begin{subequations}` und `\end{subequations}` erreichen. Beide Befehle werden im T_EX-Modus eingegeben.

⁴²`\tag` ist in Kap. 19.4 beschrieben.

Dazu ein Beispiel:

$$A = C - B \quad (17)$$

$$B = C - A \quad (17a)$$

$$C = A + B \quad (17b)$$

Um das Beispiel zu erstellen, geht man folgendermaßen vor:

1. erste Formel eingeben
2. `\addtocounter{equation}{-1}` `\begin{subequations}`
nach der ersten Formel einfügen
3. zweite Formel eingeben
4. dritte Formel eingeben
5. `\end{subequations}` nach der dritten Formel einfügen

Jede Formel, die zwischen den Befehlen `\begin` und `\end` steht, wird mit a, b, c, ... untergliedert. Bei mehrzeiligen Formeln wird jede Zeile untergliedert. Alle untergliederten Formeln gelten als *eine* nummerierte Formel. Da jede nummerierte Formel den Formelzähler **equation** um Eins erhöht, ist der Befehl `\addtocounter` notwendig um den Zähler wieder zu erniedrigen. Ansonsten würden die Formeln mit 4, 5a, 5b nummeriert werden.

Durch die Eingabe der Befehle im T_EX-Modus wird zwischen den ersten beiden Formeln automatisch etwas Leerraum erstellt. Um dies rückgängig zu machen, gibt man nach

`\begin{subequations}` im T_EX-Modus den Befehl `\vspace{-5mm}`⁴³ ein. Ist der Formelstil **Halblinks**⁴⁴ gewählt, gibt man `\vspace{-7mm}` ein.

Ein Beispiel für mehrzeilige Formeln, bei dem die Nummerierung der zweiten Zeile abgeschaltet wurde:

$$A = (B - Z)^2 = (B - Z)(B - Z) \quad (18a)$$

$$\begin{aligned} &= B^2 - ZB - BZ + Z^2 \\ &= B^2 - 2BZ + Z^2 \end{aligned} \quad (18b)$$

19.4 Benutzerdefinierte Nummerierung

Bei allen Nummerierungen werden um die Formelnummern runde Klammern gesetzt. Will man diese Klammern z. B. durch senkrechte Striche ersetzen, gibt man im L^AT_EX-Vorspann Folgendes an:

⁴³mehr zu `\vspace` siehe Kap. 22.16

⁴⁴Formelstile siehe Kap. 17

`\def\tagform@#1{\maketag@@@{|#1|}}`

Möchte man andere Zeichen haben, ersetzt man die Striche neben `#1` durch ein oder mehrere gewünschte Zeichen. Möchte man nur die Formelnummer haben, lässt man die Striche weg.

Soll statt einer fortlaufenden Nummer ein beliebiger Ausdruck in den Klammern hinter einer Formel stehen, benutzt man den Befehl `\tag{Ausdruck}`:

$$A + B = C \tag{etwas}$$

In diesem Beispiel wurde in der Formel der Befehl `\tag{etwas}` eingegeben.

Gibt man stattdessen `\tag*{etwas}` ein, verhindert der Stern die Klammern um den Ausdruck:

$$A + B = C \tag*{etwas}$$

Oftmals möchte man Formeln in folgender Art nummerieren:

`(Kapitelnummer.Formelnummer)`

Dabei soll die Formelnummer bei jedem Kapitel wieder mit Eins beginnen.

Für diesen Fall gibt es den Befehl `\numberwithin`, der nach folgendem Schema im L^AT_EX-Vorspann benutzt wird:

`\numberwithin{Zähler}{Gliederung}`

Zähler gibt an, welche Nummerierung betroffen ist. Gliederung gibt an, welche Nummer vor dem Punkt steht.

In unserem Fall gibt man demnach folgende L^AT_EX-Vorspann- oder ERT-zeile ein:

`\numberwithin{equation}{section}`

Hier das Ergebnis:

$$A + B = C \tag{19.19}$$

Möchte man z. B. Tabellen nummerieren, so dass die Nummer des Teils vor dem Punkt steht, gibt man `\numberwithin{table}{part}` ein.

Um wieder auf Standardnummerierung umzuschalten oder um diese Art der Nummerierung zu vermeiden wenn sie von der Dokumentklasse vorgegeben ist, gibt man folgenden Befehl als ERT oder als L^AT_EX-Vorspannzeile ein:

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

bzw.

`\renewcommand{\thetable}{\arabic{table}}`

Möchte man die Formelnummerierung neu starten wenn ein neuer Dokumentteil oder -abschnitt beginnt, verwendet man folgende L^AT_EX-Vorspannzeile:

`\@addtoreset{equation}{part}`

bzw.

`\@addtoreset{equation}{section}`

19.5 Nummerierung mit römischen Zahlen und Buchstaben

Des Weiteren kann man Formeln mit römischen Zahlen und lateinischen Buchstaben nummerieren. Um z. B. mit kleinen römischen Zahlen zu nummerieren, gibt man vor der Formel im \TeX -Modus den Befehl

`\renewcommand{\theequation}{\roman{equation}}`

ein. `\renewcommand` weist dem vordefinierten Befehl `\theequation` den Befehl `\roman{equation}` zu⁴⁵. `equation` ist der Formelzähler. Stellt man einem Zähler den Befehl `\the` voran, bewirkt dies, dass der Wert des Zählers als arabische Zahl ausgegeben wird. Gibt man an, dass eine Formel nummeriert wird, setzt \LaTeX intern den Befehl `\theequation` hinter die Formel. `\roman{equation}` bewirkt, dass der Formelzähler als kleine römische Zahl erscheint.

Alle Formeln nach dem `\renewcommand`-Befehl werden nun römisch nummeriert. Um auf Nummerierung mit großen römischen Zahlen umzuschalten, gibt man den Befehl erneut ein, ersetzt aber `\roman` durch `\Roman`. Für die „Nummerierung“ mit lateinischen Buchstaben gibt es den Befehl `\alph` für kleine und `\Alph` für große Buchstaben. Dabei ist zu beachten, dass maximal 26 Formeln in einem Dokument mit lateinischen Buchstaben nummeriert werden können.

$$A = \text{römisch} \quad (\text{xx})$$

$$B = \text{Römisch} \quad (\text{XXI})$$

$$C = \text{lateinisch} \quad (\text{v})$$

$$D = \text{Lateinisch} \quad (\text{W})$$

Um wieder auf Standardnummerierung umzuschalten, gibt man folgenden Befehl ein:

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

$$E = \text{arabisch} \quad (24)$$

Wie man sieht, werden die Formeln unabhängig von der Nummerierungsart durchnummeriert. Soll beim Wechsel der Nummerierungsart die Nummerierung wieder mit Eins beginnen, müssen neue Formelzähler angelegt werden. Eine Beschreibung dazu findet man in der Datei [Formelnummerierung.pdf](#).

20 Benutzerdefinierte Befehle

Die Namen von selbst definierten Befehlen und Makros dürfen nur aus lateinischen Buchstaben bestehen.

⁴⁵Der Befehl `\renewcommand` besitzt dasselbe Schema wie der in Kap. 20.1 beschriebene Befehl `\newcommand`.

20.1 Der Befehl `\newcommand`

Viele der \LaTeX -Befehle sind für den ständigen Gebrauch viel zu lang. Man kann sich deswegen mit dem Befehl `\newcommand` im \LaTeX -Vorspann neue kürzere Befehle definieren.

Das Befehlsschema von `\newcommand` lautet:

```
\newcommand{neuer Befehl}[Argumentanzahl][optionaler Wert]
               {Befehlsdefinition}
```

Dabei muss darauf geachtet werden, dass man einen neuen Befehl nimmt, der nicht schon anderweitig definiert ist. Definiert man beispielsweise für `\Leftarrow` den Befehl `\le`, erhält man eine Fehlermeldung.

Die Argumentanzahl ist eine ganze Zahl im Bereich 0-9 und gibt an, wie viele Argumente der neue Befehl haben soll. Als optionalen Wert kann man den Wert für ein optionales Argument voreinstellen. Wenn man dies macht, bedeutet das gleichzeitig, dass das *erste* Argument des neuen Befehls ein optionales ist.

Es folgen einige Beispiele:

- Will man für `\Longrightarrow` den Befehl `\gr` definieren, lautet die \LaTeX -Vorspannzeile:

```
\newcommand{\gr}{\Longrightarrow}
```

- Um für den Befehl `\underline` den Befehl `\us` zu definieren, muss das Argument (was unterstrichen werden soll) berücksichtigt werden. Dazu sieht die Vorspannzeile wie folgt aus:

```
\newcommand{\us}[1]{\underline{#1}}
```

Das Zeichen `#` fungiert als Argumentplatzhalter, die `1` dahinter gibt an, dass es der Platzhalter für das erste Argument ist.

- Für `\framebox` kann man z. B. den Befehl `\fb` definieren:

```
\newcommand{\fb}[3]{\framebox#1#2{ $#3$ }}
```

Die beiden Dollarzeichen erstellen hierbei die für `\framebox` geforderte weitere Formel, siehe Kap. 9.1.

- Um einen neuen Befehl für `\fcolorbox` zu erstellen, bei dem die Farbe der Box nicht mehr angegeben werden muss, definiert man das Argument für die Farbe als optional:

```
\newcommand{\cb}[3][white]{\fcolorbox{#2}{#1}{ $#3$ }}
```

Gibt man bei der Verwendung von `\cb` die Farbe nicht an, wird die voreingestellte Farbe **white** verwendet.

Ein Test der neu definierten Befehle:

Befehl	Ergebnis
$A \backslash \text{gr} _ B$	$A \Longrightarrow B$
$\backslash \text{us}\{ABcd$	\underline{ABcd}
$\backslash \text{fb}\{[2cm] \rightarrow \{ \rightarrow \{ \backslash \text{int} _ A=B$	$\int A = B$
$\backslash \text{cb}\{\text{red} \rightarrow \{ \backslash \text{int} _ A=B$	$\int A = B$
$\backslash \text{cb}[\text{green}]\{\text{red} \rightarrow \{ \backslash \text{int} _ A=B$	$\int A = B$

20.2 Mathe-Makros

Besonders praktisch sind eigene Befehle für komplizierte Ausdrücke. Hat man es in einem Dokument z. B. häufiger mit quadratischen Gleichungen zu tun, tritt immer dieselbe Lösung in verschiedenen Versionen auf. Die allgemeine Form einer quadratischen Gleichung ist

$$0 = \lambda^2 + p\lambda + q$$

Die allgemeine Form der Lösung lautet

$$\lambda_{1,2} = -\frac{p}{2} \pm \sqrt{\frac{p^2}{4} - q}$$

Um für die Lösungsformel einen Befehl zu definieren, bei dem man nur noch die drei Parameter λ , p und q eingeben muss, lautet die \LaTeX -Vorspannzeile

```
\newcommand{\qG}[3]{#1_{#2}=-\frac{#2}{#3}\pm
\sqrt{\frac{#2^2}{#4}-#3}}
```

Um damit die Lösungsformel zu erstellen, gibt man in einer Formel den Befehl $\backslash \text{qG}\{\backslash \text{lambda} \rightarrow \{ \text{p} \rightarrow \{ \text{q}$ ein.

Allerdings ist die Definition des neuen Befehls nicht gerade einfach. Denn einerseits muss man die Schemata aller verwendeten \LaTeX -Befehle kennen, z. B. dass ein Bruch in \LaTeX mit $\backslash \text{frac}\{\text{Zähler}\}\{\text{Nenner}\}$ eingegeben wird, andererseits verliert man leicht den Überblick und vergisst eventuell eine geschweifte Klammer. Um diesen Problemen aus dem Weg zu gehen, gibt es in $\text{L\kern-.1em X\kern-.1em T\kern-.1em E\kern-.1em X}$ die Möglichkeit, Mathe-Makros statt dem Befehl $\backslash \text{newcommand}$ zu verwenden.

Ein Mathe-Makro wird erstellt, indem man den Befehl

math-macro Name Argumentanzahl

in den so genannten Kommandopuffer eingibt. Der Kommandopuffer wird am unteren Rand des $\text{L\kern-.1em X\kern-.1em T\kern-.1em E\kern-.1em X}$ -Fensters eingeblendet, wenn man mit der rechten Maustaste auf die

Menüsymbolleiste klickt und *Command-Buffer* auswählt.

Der Name im Makro-Befehl ist dabei der Name des neu definierten Befehls, der keine Ziffer enthalten darf. Die Argumentanzahl ist eine Zahl im Bereich 1-9. Möchte man einen neuen Befehl ohne Argument definieren, gibt man keine Argumentanzahl an.

Wir haben drei Argumente und nennen das Makro „qG“, so dass der Befehl lautet:

math-macro_qG_3

Drückt man nach dem Befehl **Enter**, erscheint im Dokument folgende Makrodefinitionsbox:

Makro: qG:

In das erste blaue Kästchen gibt man die alten Befehle wie in eine normale Formel ein. Einen Argumentplatzhalter fügt man mit dem Befehl `\#Argumentnummer` ein, z. B. `\#1`. In LyX wird ein Platzhalter rot dargestellt.

In das zweite Kästchen kann man eingeben, in welcher Form das Makro in LyX angezeigt werden soll. Normalerweise will man es so sehen, wie es definiert wurde. Dazu lässt man das zweite Kästchen leer. Hat man hingegen ein Makro für einen Ausdruck erstellt, der auf dem Bildschirm viel Platz einnimmt, kann man in das zweite Kästchen z. B.

qG: \#1 , \#2 , \#3

eingeben. Für das Makro werden dann in LyX nur die drei Argumente mit dem davor stehenden Makronamen angezeigt, was für mehr Überblick im Dokument sorgt. Im Ausdruck erscheint die Formel so, wie sie im ersten Kästchen definiert wurde.

Um ein Makro zu verwenden, gibt man in eine Formel den Makronamen als Befehl ein, in unserem Fall „`\qG`“. Es erscheint dann das Makro und gegebenenfalls die Argumente darunter. In unserem Fall erscheint:

$$\text{qG}_{1,2} = -\frac{\text{\#2}}{2} \pm \sqrt{\frac{\text{\#2}}{4} - \text{\#3}}$$

\#1:
 \#2:
 \#3:

In die blauen Kästchen werden die Argumente eingegeben. Verlässt man mit dem Cursor das Makro, werden die Argumente in das Makro eingesetzt. Um Argumente zu ändern, setzt man den Cursor in der Formel vor oder nach das Makro und drückt die rechte bzw. linke Pfeiltaste. Es erscheinen nun wieder die Argumente unterhalb der Formel.

Ein Mathe-Makro wird beim Exportieren des Dokuments von LyX in einen `\newcommand`-Befehl umgewandelt, jedoch ist ein Makro im Vergleich zum direkten Befehl einfacher zu erstellen. Außerdem sieht man in LyX bereits das Ergebnis. Wenn

man z. B. eine Makrodefinition nachträglich ändert, werden die Makroformeln automatisch angepasst. Hier ein Beispiel unseres Makros mit den Argumenten x , $\ln(x)$ und B :

$$x_{1,2} = -\frac{\ln(x)}{2} \pm \sqrt{\frac{\ln(x)^2}{4} - B}$$

Mathe-Makros können auch direkt aus einem `\newcommand`-Befehl erstellt werden. Schreibt man z. B. den Befehl

`\newcommand{\larrow}[2]{\xleftarrow[#2]{#1}}`

markiert ihn komplett mit der Maus und verwendet dann das Tastenkürzel **Strg-m**, wird der Befehl in ein Mathe-Makro umgewandelt. Bei dieser Methode muss man jedoch aufpassen, dass der `\newcommand`-Befehl richtig geschrieben wurde, ansonsten wird ein fehlerhaftes Mathe-Makro erstellt, dass später \LaTeX -Fehler hervorruft.

Bei Makros muss beachtet werden, dass der aus einem Makro erzeugte `\newcommand`-Befehl nicht in den \LaTeX -Vorspann gesetzt wird. Deswegen kann man ein Makro nur in Formeln verwenden, die sich im Dokument unterhalb der Makrodefinitionsbox befinden. Der Makroname und die Argumentanzahl können nicht nachträglich geändert werden, und in \LaTeX ist nicht ersichtlich, wie viele Argumente das Makro besitzt. Des Weiteren werden optionale Argumente in Makros noch nicht unterstützt. Weitere Formeln in einer Makrodefinition werden fehlerhaft behandelt⁴⁶. Dadurch kann das dritte Beispiel aus dem vorigen Kapitel nicht als Makro geschrieben werden.

21 Diagramme

Um Diagramme zu erstellen, wird das \LaTeX -Paket `amscd`⁴⁷ benötigt. Es wird im \LaTeX -Vorspann mit der Zeile

`\usepackage{amscd}`

geladen.

Diagramme stellen Beziehungen dar und sehen folgendermaßen aus:

$$\begin{array}{ccccc} A & \longrightarrow & B & \longrightarrow & C \\ \uparrow & & & & \downarrow \\ F & \longleftarrow & E & \longleftarrow & D \end{array}$$

⁴⁶ [\$\text{\LaTeX}\$ -Fehlernummer #1452](#)

⁴⁷`amscd` ist Teil jeder \LaTeX -Standardinstallation.

Um sie zu erstellen, gibt man in einer Formel den Befehl `\CD` ein. Es erscheint ein blaues Kästchen mit zwei gestrichelten Linien, in das man die weiteren Befehle eingibt. Mit **Strg-Enter** wird eine neue Zeile erstellt. Horizontale Beziehungen werden in ungerade Zeilen, vertikale in gerade Zeilen der Formel eingegeben.

Zum Erstellen der Beziehungen gibt es folgende Befehle:

- `@<<<` erstellt einen Linkspfeil, `@>>>` einen Rechtspfeil und `@=` ein langes Gleichheitszeichen.
- `@AAA` erstellt einen Pfeil nach oben, `@VVV` einen Pfeil nach unten und `@|` ein senkrechtes Gleichheitszeichen
- `@.` dient als Platzhalter für nicht vorhandene Beziehungen

Alle Pfeile sind wie folgt beschriftbar:

- Wird bei horizontalen Pfeilen zwischen dem ersten und zweiten `<` bzw. `>` Text eingegeben, erscheint er über dem Pfeil. Wird der Text zwischen dem zweiten und dritten `<` bzw. `>` eingegeben, erscheint er unter dem Pfeil.
- Wird bei vertikalen Pfeilen Text zwischen erstem und zweitem `A` bzw. `V` eingegeben, erscheint er links neben dem Pfeil. Bei Eingabe zwischen zweitem und drittem `A` bzw. `V` erscheint er rechts daneben. Enthält der Text ein `A` oder `V`, setzt man ihn in eine \LaTeX -Klammer.

Als Beispiel ein Diagramm mit allen möglichen Beziehungen:

$$\begin{array}{ccccccc}
 A & \xrightarrow{j} & B & \xrightarrow[k]{} & C & \xlongequal{\quad} & F \\
 m \uparrow & & & & \downarrow V & & \parallel \\
 D & \xleftarrow[j]{} & E & \xrightarrow[k]{} & F & \xlongequal{\quad} & C
 \end{array}$$

Der Befehl dazu lautet:

```

\CD_A@>j>>B@>>k>C@=F Strg-Enter
@AmAA@.@VV\{V\to V@| Strg-Enter
D@<<j<E@>k>>F@=C

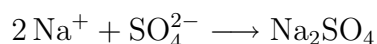
```

22 Tipps⁴⁸

22.1 Chemische Symbole und Reaktionsgleichungen

Ein Beispielttext aus der Chemie:

Das SO_4^{2-} -Ion verbindet sich mit zwei Na^+ -Ionen zu Natriumsulfat (Na_2SO_4).
Die Reaktionsgleichung dazu lautet:



Diese Reaktionsgleichung kann direkt als Formel erstellt werden. Damit die Symbole nicht kursiv erscheinen, markiert man sämtliche Zeichen in der Formel und wandelt diese mit der Tastenkombination **Alt-z r**⁴⁹ in eine aufrechte Form um.

22.2 Negative Zahlen

Ein häufiges Problem stellen negative Zahlen im Formelmodus dar, denn das Minuszeichen vor der Ziffer wird mit der gleichen Länge wie das Minuszeichen der gleichnamigen Operation dargestellt. Schreibt man dagegen eine negative Zahl im normalen Text, erscheint das Minuszeichen korrekt.

Wandelt man also das Minuszeichen in mathematischen Text um, tritt das Problem nicht mehr auf.

Ein Beispiel zur Veranschaulichung des Problems:

normaler Text:	$x = -2$
Formelmodus:	$x = -2$
Lösung:	$x = -2$

22.3 Komma als Dezimaltrennzeichen

In \LaTeX fungiert das Komma in einer Formel, gemäß englischer Konvention, als Kolonnentrennzeichen. Bei Eingabe eines Kommas in eine Formel wird demnach automatisch hinter dem Komma ein Leerraum eingefügt.

Um dies zu verhindern, markiert man das Komma und wandelt es in mathematischen Text um (Tastenkürzel **Alt-m m**).

Um hingegen alle Formelkommata des Dokuments als Dezimaltrennzeichen zu setzen, lädt man nur die Datei **icomma.sty**⁵⁰ mit der \LaTeX -Vorspannzeile

```
\usepackage{icomma}
```

⁴⁸Weitere gute Tipps gibt es unter [3].

⁴⁹Schriftstile siehe Kap. 11.1

⁵⁰**icomma** ist Teil des \LaTeX -Paketes **was**.

22.4 Physikalische Vektoren

Vordefinierte Vektoren bietet das Paket **braket**⁵¹, das mit der \LaTeX -Vorspannzeile `\usepackage{braket}`

geladen wird.

Folgende Befehle sind definiert:

Befehl	Ergebnis
<code>\Bra{\psi}</code>	$\langle\psi $
<code>\Ket{\psi}</code>	$ \psi\rangle$
<code>\Braket{\psi \phi}</code>	$\langle\psi \phi\rangle$

Der Befehl `\Braket` stellt sicher, dass alle senkrechten Striche in der Größe der umgebenden Klammern gesetzt werden:

$$\left\langle\phi\left|J=\frac{3}{2},M_J\right.\right\rangle$$

22.5 Selbst definierte Brüche

Um eigene Befehle für Brüche zu definieren, verwendet man den Befehl `\genfrac`. Dieser wird nach folgendem Schema verwendet:

`\genfrac{linke Klammer}{rechte Klammer}{Bruchstrichdicke}{Ansichtsstil}{Zähler}{Nenner}`

Der Ansichtsstil wird mit einer ganzen Zahl im Bereich 0-3 angegeben.

Nummer	Ansichtsstil
0	abgesetzte Formel
1	eingebettete Formel
2	klein
3	winzig

Wird nichts für den Ansichtsstil angegeben, passt sich die Größe wie bei `\frac` der Umgebung an.

Wird keine Bruchstrichdicke angegeben, wird der voreingestellte Wert von 0.4 pt verwendet.

Zum Beispiel sind die Befehle `\dffrac` und `\tbinom` folgendermaßen definiert:

⁵¹**braket** ist Teil jeder \LaTeX -Standardinstallation.


```
\newcommand{\dfrac}[2]{\genfrac{}{}{0}{#1}{#2}}
\newcommand{\tbinom}[2]{\genfrac{({}{)}{0pt}{1}{#1}{#2}}
```

Um einen Bruch zu definieren, bei dem man die Bruchstrichdicke als optionales Argument angeben kann, gibt man im L^AT_EX-Vorspann folgenden Befehl ein:

```
\newcommand{\fracS}[3]{\genfrac{}{}{#1}{#2}{#3}}
```

Dazu ein Test:

Befehl	$\fracS[1mm]{A \rightarrow}{B}$	$\fracS[5mm]{A \rightarrow}{B}$
Ergebnis	$\frac{A}{B}$	$\frac{A}{B}$

Wie man sieht, entspricht der Abstand des Zählers und Nenners vom Strich in etwa der dreifachen Strichdicke.

Um einen Bruch mit einem schrägen Bruchstrich zu erstellen, gibt man im L^AT_EX-Vorspann folgenden Befehl ein:

```
\newcommand{\textfrac}[2]{%
\raisebox{1mm}{\scriptsize #1}}%
\hspace{-0.5mm}%
\raisebox{0.2mm}{\small /}%
\hspace{-0.5mm}%
\raisebox{-0.6mm}{\scriptsize #2}}%
```

Der damit definierte Befehl `\textfrac` erzeugt Brüche, die sich besonders für Text eignen. So erzeugt z. B. der Befehl `\textfrac{16 \rightarrow}{3}` den Bruch $16/3$.

`\textfrac` kann auch im T_EX-Modus verwendet werden.

22.6 Durchgestrichene Formeln

Um Formeln oder Formelteile durchzustreichen, muss das L^AT_EX-Paket **cancel**⁵² mit der L^AT_EX-Vorspannzeile

```
\usepackage{cancel}
```

geladen werden.

⁵²**cancel** ist Teil jeder L^AT_EX-Standardinstallation.

Es gibt vier verschiedene Arten durchzustreichen:

Befehl	Ergebnis
<code>\cancel{\int A=B}</code>	$\int A=\cancel{B}$
<code>\bcancel{\int A=B}</code>	$\int A=\cancel{B}$
<code>\xcancel{\int A=B}</code>	$\int A=\cancel{B}$
<code>\cancelto{1\rightarrow}\{\int A=B\}</code>	$\int A=\cancel{B}^1$

`\cancelto` eignet sich besonders, wenn man das Kürzen innerhalb einer Formel darstellen will:

$$\frac{(x_0 + bB)^2}{(1 + b^2)^2} = \frac{x_0^2 + B^2 - r_g^2}{1 + \cancel{b^2}}$$

22.7 Formeln in Überschriften

Benutzt man Formeln in Überschriften, ist auf Folgendes zu achten:

- Verwendet man das L^AT_EX-Paket **hyperref**, werden automatisch PDF-Lesezeichen für jede Überschrift aus dem Inhaltsverzeichnis erzeugt. Enthält eine Überschrift Formeln, werden diese im Lesezeichentext falsch dargestellt. Denn Formeln in Lesezeichen verstoßen gegen die PDF-Konventionen.
- Es ist nicht üblich im Inhaltsverzeichnis Formeln zu verwenden.

Beide Probleme lassen sich lösen, indem man am Ende der Überschrift einen Kurztitel über das Menü **Einfügen** \triangleright **Kurztitel** einfügt. Kurztitel werden als Alternative für mehrzeilige Überschriften verwendet, um das Inhaltsverzeichnis übersichtlich zu halten. Nur der Kurztitel erscheint im Inhaltsverzeichnis und damit auch im PDF-Lesezeichen.

Möchte man im Inhaltsverzeichnis eine Formel verwenden, benutzt aber das Paket **hyperref**, verwendet man im T_EX-Modus den Befehl

`\texorpdfstring{Zeichen}{Alternativzeichen}`

Zeichen sind die Überschriftsteile, die nicht im PDF-Lesezeichen erscheinen sollen. Das können außer Formeln auch Fußnoten oder Querverweise sein. Als Alternativzeichen kann man angeben, was stattdessen im Lesezeichen erscheinen soll. Möchte man keine Alternative angeben, schreibt man nur die geschweiften Klammern.

Es folgen zwei Beispielüberschriften:

22.7.1 Überschrift ohne Formel im Inhaltsverzeichnis $\sqrt{-1} = i$

22.7.2 Überschrift mit Formel im Inhaltsverzeichnis $\sqrt{-1} = i$

In der ersten Überschrift wurde ein Kurztitel verwendet. In der zweiten `\texorpdfstring`.

Damit die Formeln so wie der Rest der Überschrift formatiert werden, wurde die komplette Überschrift in eine **boldmath-Umgebung**⁵³ gesetzt.

22.8 Formeln im mehrspaltigen Text

Formeln im mehrspaltigen Text sind oftmals zu breit für eine Spalte und müssen daher über die gesamte Seitenbreite gesetzt werden. Dazu verwendet man das Paket **multicol**⁵⁴, das mit der Vorspannzeile

`\usepackage{multicol}`

geladen wird. Dabei ist zu beachten, dass im Menü **Dokument** \triangleright **Einstellungen** unter **Textformat** *nicht* **Zweispaltiges Dokument** ausgewählt sein darf.

Vor den mehrspaltigen Text schreibt man im T_EX-Modus den Befehl

`\begin{multicols}{Spaltenanzahl}`

wobei die Spaltenanzahl eine Zahl im Bereich 2 - 10 ist. Vor der Formel beendet man den mehrspaltigen Text mit dem Befehl

`\end{multicols}`

der im T_EX-Modus eingegeben wird.

Durch den Befehl wird vor der Formel automatisch etwas Leerraum erstellt. Um dies rückgängig zu machen, schreibt man im T_EX-Modus vor die Formel den Befehl

`\vspace{-6mm}`⁵⁵. Ist der Formelstil **Halblinks**⁵⁶ gewählt, schreibt man stattdessen `\vspace{-9mm}`.

⁵³siehe Kap. 11.2

⁵⁴**multicol** ist Teil jeder L^AT_EX-Standardinstallation.

⁵⁵mehr zu `\vspace` siehe Kap. 22.16

⁵⁶Formelstile siehe Kap. 17

Als Beispiel ein mehrspaltiger Text mit einer abgesetzten Formel:

Das Spektrum wird fouriertransformiert. Die Fouriertransformation wird verwendet, um die überlagerten Signale (Netzwerk, Lösungsmittel) zu trennen. Nachdem wir die Phasenverschiebung bestimmen konnten, interessiert uns nun das Aussehen des Ausgangssignals. Im Experiment haben wir es mit sehr vielen Teilchen zu tun, so dass man über alle Phasen integrieren muss. Sei nun S unser normiertes Ausgangssignal und P die Phasenverteilungsfunktion, so ergibt sich die Beziehung

$$S(t) = S_0(t) \int_{-\infty}^{\infty} P(\phi, t) e^{i\phi} d\phi \quad (1)$$

wobei S_0 das Signal ohne Gradient ist und die Normierungsbedingung $\int_{-\infty}^{\infty} P(\phi, t) d\phi = 1$ gilt. Nun dürfen wir aber nicht den Relaxationsprozess außer Acht lassen. Direkt nach dem $\pi/2$ -rf-Puls beginnt sich die Magnetisierung zu entfokussieren, wodurch sich das Signal zusätzlich abschwächt. Diese Abschwächung verläuft exponentiell in Abhängigkeit der so genannten T_2 -Zeit.

22.9 Formeln mit Beschreibung der Variablen

Möchte man wie in (2) die verwendeten n Variablen innerhalb einer Formel beschreiben, verwendet man eine Matrix $(2, n)$ mit links ausgerichteten Spalten⁵⁷. Um die Beschreibung kleiner zu setzen, gibt man vor der Matrix z. B. den Befehl `\footnotesize`⁵⁸ ein.

Wird der Formelstil **Halblinks**⁵⁹ verwendet, fügt man vor und nach der Matrix ein `\hfill`⁶⁰ ein, damit der Abstand der Matrix von der Funktion und vom Seitenrand gleich ist.

Ist der Stil **Zentriert** gewählt, nutzt man die in Kap. 18.2.3 beschriebene Möglichkeit, Formeln halblinks auszurichten. (2) besteht aus fünf Spalten, wobei in den ersten beiden die Funktion, in der dritten die Matrix und in der letzten die leere L^AT_EX-Klammer steht.

$F_A = \rho \cdot V \cdot g$		ρ Dichte V Volumen g Fallbeschleunigung	(2)
------------------------------	--	--	-----

⁵⁷Matrizen siehe Kap. 4

⁵⁸Schriftgrößen siehe Kap. 11.3

⁵⁹Formelstile siehe Kap. 17

⁶⁰`\hfill` wirkt nur in Formeln mit dem Stil **Halblinks**, siehe Kap. 8.2.

22.10 Aufrechte kleine griechische Buchstaben

Die meisten Schriften stellen nur kursive kleine griechische Buchstaben zur Verfügung. Für Symbole der Elementarteilchen, wie Pionen und Neutrinos werden jedoch aufrechte Buchstaben benötigt. Die Datei **upgreek.sty**⁶¹, die mit der L^AT_EX-Vorspannzeile

```
\usepackage{upgreek}
```

geladen wird, stellt diese Buchstaben zur Verfügung. Sie werden erzeugt, wenn der Befehl für einen griechischen Buchstaben mit **up** begonnen wird. So ergibt z. B. der Befehl `\uptau`: τ

Damit kann man Elementarteilchenreaktionen erstellen:

$$\pi^+ \rightarrow \mu^+ + \nu_\mu$$

Die aufrechten Buchstaben sind fetter und breiter als die kursiven, da sie speziell für Elementarteilchensymbole entworfen wurden. Für Einheiten wie μm , sollte man daher den Befehl `\textmu` verwenden, siehe Kap. 22.12.

22.11 Textzeichen in Formeln

In einigen Fällen möchte man Textzeichen direkt in eine Formel eingeben. Benötigt man z. B. den mittigen Punkt \cdot in Formeln wie $\nu = 5 \cdot 10^5 \text{ Hz}$, müsste man dafür den Befehl `\cdot` eingeben⁶², da das Zeichen in allen Kodierungen als Textzeichen definiert ist. Man kann jedoch die Kodierung mit folgender L^AT_EX-Vorspannzeile ändern:

```
\DeclareInputText{183}{\ifmmode\cdot\else\textperiodcentered\fi}
```

Die Zeichenkodierung (Menü **Dokument** \triangleright **Einstellungen** \triangleright **Sprache**) legt fest, welches Zeichen beim Druck einer bestimmten Taste erscheint. Wenn man die Taste für das Zeichen `'.'` drückt, wird intern der Befehl `\textperiodcentered` verwendet. Dieser Befehl ist in einer Formel jedoch nicht verfügbar, so dass es zu L^AT_EX-Fehlern kommen würde. Durch die geänderte Kodierung wird hingegen automatisch der richtige Befehl gewählt, je nachdem ob das Zeichen in eine Formel eingegeben wurde oder nicht.

Die Kodierung der einzelnen Zeichen ist in Definitionsdateien abgespeichert. Z. B. ist die Kodierung **latin9** in der Datei **latin9.def** definiert, die sich im Installationsverzeichnis von L^AT_EX befindet. Man sollte die Kodierung nur über den L^AT_EX-Vorspann ändern und die Definitionsdateien unverändert lassen. Ansonsten können eigene Dateien von anderen L^AT_EX-Nutzern nicht ohne Weiteres bearbeitet werden.

In diesem Dokument wurden außer dem mittigen Punkt noch das Gradzeichen $^\circ$ und das gelegte Kreuz \times mit folgenden L^AT_EX-Vorspannzeilen so definiert, dass sie direkt in eine Formel eingegeben werden können:

⁶¹**upgreek** ist Teil des L^AT_EX-Paketes **was**.

⁶²siehe Kap. 10.3

```
\DeclareInputText{176}{\ifmmode^\circ\else\textdegree\fi}
\DeclareInputText{215}{\ifmmode\times\else\texttimes\fi}
```

22.12 Symbole aus dem textcomp-Paket

Um die folgenden Symbole nutzen zu können, muss das L^AT_EX-Paket **textcomp**⁶³ mit der L^AT_EX-Vorspannzeile

```
\usepackage{textcomp}
```

geladen werden.

Alle textcomp-Befehle sind nur im T_EX-Modus und im mathematischen Textmodus verfügbar.

Befehl	Ergebnis	Befehl	Ergebnis
<code>\textcent</code>	¢	<code>\textdied</code>	†
<code>\texteuro</code>	€	<code>\textmarried</code>	∞
<code>\textperthousand</code>	‰	<code>\textnumero</code>	№
<code>\textmu</code>	μ	<code>\textonequarter</code>	$\frac{1}{4}$
<code>\texttimes</code>	×	<code>\textonehalf</code>	$\frac{1}{2}$
<code>\texttrademark</code>	™	<code>\textthreequarters</code>	$\frac{3}{4}$
<code>\textohm</code>	Ω	<code>\textbrokenbar</code>	

Bei den Bruch-Symbolen ist der Abstand Zahl - Bruchstrich geringer als bei einer Formel. Symbol: $\frac{3}{4}$ Formel: $\frac{3}{4}$

Bei manchen Schriften werden die Bruch-Symbole auch mit einem schrägen Bruchstrich angezeigt, in etwa so: $\frac{3}{4}$

Moderne Schriften, wie z. B. die Schrift **lm** (latin modern), besitzen ein besser aussehendes Eurozeichen. Dies wird verwendet, wenn **textcomp** mit der Option **euro** geladen wird.

22.13 Textformatierung

Alle in diesem Kapitel aufgeführten Befehle sind nur im T_EX-Modus und im mathematischen Textmodus verfügbar.

Außer den Befehlen der in diesem Kapitel vorgestellten Pakete **soul** und **ulem** können auch die Befehle `\cancel`, `\bcancel` und `\xcancel` des in Kap. 22.6 beschriebenen Pakets **cancel** zum Durchstreichen von Text verwendet werden.

⁶³**textcomp** ist Teil jeder L^AT_EX-Standardinstallation.

22.13.1 Das Paket soul

soul wird mit der L^AT_EX-Vorspannzeile

`\usepackage{soul}`

geladen und stellt folgende Befehle zur Verfügung:

Befehl	Ergebnis
<code>\so{Gesperrt}</code>	G e s p e r r t
<code>\hl{Farbig unterlegt}</code>	Farbig unterlegt
<code>\st{Durchgestrichen}</code>	Durchgestrichen
<code>\ul{Unterstrichen}</code>	<u>Unterstrichen</u>

Text sperren heißt nach Duden, Text durch Leerraum zwischen jedem Buchstaben optisch hervorzuheben. Durch den in der Dokumentation⁶⁴ von **soul** beschriebenen Befehl `\sodef`, kann der Leerraum individuell angepasst werden.

Der Befehl `\hl` unterlegt Text nur farbig, wenn im L^AT_EX-Vorspann das in Kap. 9.3 beschriebene Paket **color** geladen ist. Ansonsten verhält sich `\hl` wie der Befehl `\ul`. Die voreingestellte Hintergrundfarbe **yellow** kann mit dem Befehl

`\sethlcolor{Farbe}`

geändert werden, wobei alle in Kap. 9.3 beschriebenen Farben zulässig sind. `\sethlcolor` kann im L^AT_EX-Vorspann als auch im T_EX-Modus verwendet werden. Analog zu `\sethlcolor` gibt es die Befehle

`\setstcolor` und `\setulcolor`

um Text farbig ~~durchzustreichen~~ und zu unterstreichen.

Unterstreicht man Text mit dem Befehl `\ul` anstatt über das Menü **Bearbeiten**▷**Textstil**, hat das den Vorteil, dass Buchstabenteile unterhalb der Grundlinie nicht durchgestrichen werden:

mit `\ul`: Kupplung über Menü **Bearbeiten**▷**Textstil**: Kupplung

Soll das L^AT_EX-Menü standardmäßig mit `\ul` unterstreichen, fügt man im L^AT_EX-Vorspann die Zeile

`\renewcommand{\underbar}[1]{\ul{#1}}`

ein.

Die Dicke und Position des Unter- bzw. Durchstrichs kann mit dem Befehl `\setul` eingestellt werden. Mehr dazu steht in der Dokumentation von **soul**.

⁶⁴Die Dokumentation ist in der Datei **soul.dvi** zu finden.

22.13.2 Das Paket ulem

ulem ist Teil jeder L^AT_EX-Standardinstallation und wird im L^AT_EX-Vorspann mit der Zeile

```
\usepackage[normalem]{ulem}
```

geladen. Die Option **normalem** verhindert, dass das Paket die Erscheinungsform von hervorgehobenem Text undefiniert. Anstatt kursiv würde dieser aufrecht und unterstrichen erscheinen.

ulem stellt die folgenden Befehle bereit:

Befehl	Ergebnis
<code>\sout{Falsch}</code>	<u>Falsch</u>
<code>\xout{Gelöscht}</code>	Gelöscht
<code>\uwave{Wellig}</code>	<u>Wellig</u>
<code>\uuline{Doppelt}</code>	<u><u>Doppelt</u></u>

Im Gegensatz zum Paket **soul**, kann mit **ulem** formatierter Text nicht richtig umgebrochen werden:

~~Wörter können nicht getrennt werden, ragen deswegen über das Zeilenende hinaus:
Langestestwort.~~

ulem eignet sich daher nur für einzelne Wörter.

22.14 Anführungszeichen

Im Allgemeinen wird der Stil der Anführungszeichen im Menü **Dokument** ▸ **Einstellungen** unter **Sprache** festgelegt. Möchte man in einem Dokument zwei verschiedene Stile verwenden, kann man die gewünschten Zeichen auch im T_EX-Modus eingeben.

Befehl	Ergebnis
<code>\flq</code>	<
<code>\flqq</code>	«
<code>\glq</code>	,
<code>\glqq</code>	„

Befehl	Ergebnis
<code>\frq</code>	>
<code>\frqq</code>	»
<code>\grq</code>	‘
<code>\grqq</code>	“
<code>\dq</code>	”

Die Zeichen < > « und » sind in der Schrift **ae** nicht verfügbar. Um die Zeichen « und » trotzdem mit **ae** nutzen zu können, installiert man das Paket **aeguill** und lädt es mit der L^AT_EX-Vorspannzeile

```
\usepackage[ec]{aeguill}
```

Die Anführungszeichen „ und “ sind in der Schrift **ae** nicht in der Stilart **Schreibmaschine** verfügbar.

22.15 Textboxen

Um die Befehle dieses Kapitels nutzen zu können, muss das Paket **graphicx**⁶⁵ mit der \LaTeX -Vorspannzeile

```
\usepackage{graphicx}
```

geladen werden⁶⁶.

Achtung! Die meisten DVI-Programme können keine rotierten oder skalierten Texte darstellen. In der PDF- oder PostScript-Ausgabe wird aber alles korrekt angezeigt.


22.15.1 Rotierter Text

Um Text zu drehen, verwendet man den Befehl `\rotatebox` im \TeX -Modus nach folgendem Schema:

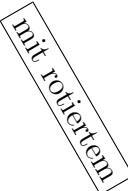
```
\rotatebox[Drehpunkt]{Winkel}{Boxinhalt}
```

Der Drehpunkt wird mit **origin=Position** angegeben. Folgende Positionen sind möglich: *c* (mittig), *l* (links), *r* (rechts), *b* (unten), *t* (oben), sowie sinnvolle Kombinationen aus den vier Grundpositionen. Z. B. bedeutet *lt*, dass sich der Drehpunkt links unten befindet. Wird kein Drehpunkt angegeben, wird die Position *l* verwendet. Winkel gibt den Drehwinkel in Grad an und darf auch negativ sein. Gedreht wird gegen den Uhrzeigersinn.

Im folgenden Beispiel wurde vor dem Text der Befehl `\rotatebox[origin=c]{60}{` im \TeX -Modus eingegeben. Nach dem Text wurde die schließende Klammer `}` wiederum im \TeX -Modus eingegeben.

Dies ist eine Zeile  Text.

Der Boxinhalt kann auch eine andere Box sein:

Dies ist eine Zeile  umrandeten Text.

⁶⁵Eine genaue Beschreibung von **graphicx** findet man in der Datei **grfguide.ps**, die wie das Paket Teil jeder \LaTeX -Standardinstallation ist.

⁶⁶Wird irgendwo im Dokument eine Grafik eingefügt, lädt \LaTeX das Paket **graphicx** automatisch. Dadurch kann es vorkommen, dass das Paket doppelt geladen wird. Dies ruft aber keine Fehler hervor.

22.15.2 Skalierter Text

Um Text zu skalieren, kann man die Befehle `\scalebox` und `\resizebox` im \TeX -Modus verwenden.

`\scalebox` wird nach folgendem Schema verwendet:

`\scalebox{horizontal}[vertikal]{Boxinhalt}`

Horizontal und vertikal geben die jeweiligen Skalierungsfaktoren an. Wird kein vertikaler Skalierungsfaktor angegeben, wird dafür der horizontale verwendet.

Z. B. ergibt der Befehl `\scalebox{2}{Hallo}` ein im Vergleich zur Dokumentschriftgröße doppelt so großes **Hallo**.

`\scalebox{2}[1]{Hallo}` verzerrt hingegen das **Hallo**.

Die Skalierungsfaktoren dürfen auch negativ sein. In diesem Fall wird der Boxinhalt gespiegelt.

Daher lässt sich mit dem Befehl `\scalebox{-1}[1]{Hallo}` Spiegelschrift erstellen: **oHsH**

`\scalebox{1}[-1]{Hallo}` spiegelt das **Hallo** an der Grundlinie.

Zu `\scalebox{-1}[1]{Boxinhalt}` gibt es den äquivalenten Befehl `\reflectbox{Boxinhalt}`.

`\resizebox` verwendet man um den Boxinhalt auf eine bestimmte Höhe bzw. Breite zu skalieren. Das Befehlsschema ist folgendes:

`\resizebox{Breite}{Höhe}{Boxinhalt}`

Gibt man für eines der beiden Argumente ein Ausrufezeichen ein, wird die Größe so gewählt, dass der Inhalt nicht verzerrt wird.

Der Befehl `\resizebox{3cm}{2cm}{Hallo}` ergibt: **Hallo**
Der Befehl `\resizebox{3cm}{!}{Hallo}` ergibt: **Hallo**

Wurde in einem Argument von `\scalebox` oder `\resizebox` eine Null angegeben, entstehen beim Exportieren zwar keine \LaTeX -Fehler, die erzeugten Dateien können jedoch nicht oder nur fehlerhaft angezeigt werden.

22.15.3 Spezielle Anwendungen

Alle Boxen können beliebig miteinander kombiniert werden. So ergibt z. B. der Befehl `\rotatebox[origin=c]{-45}{\resizebox{3cm}{!}{\reflectbox{Hallo}}}`:

OLSH

Des Weiteren kann man auch eingebettete Formeln drehen und skalieren. Dabei dürfen die Boxbefehle aber nicht in die Formel eingegeben werden, sondern müssen wie bei Text im \TeX -Modus verwendet werden.

Dies ist eine Zeile mit $\int A dx = B$ einer rotierten Formel.

Hat man den Formelstil **Halblinks**⁶⁷ gewählt, kann man auch abgesetzte Formeln skalieren.

22.16 Vertikaler Leerraum im Text

In \LyX kann über das Menü **Einfügen** \triangleright **Besondere Formatierung** \triangleright **Vertikaler Abstand** vertikaler Leerraum eingefügt werden. Der Leerraum *Standard* entspricht entweder der Größe *Mittel* oder dem Abstand, der im Menü **Dokument** \triangleright **Einstellungen** \triangleright **Textformat** zum Trennen von Absätzen eingestellt ist. Die verschiedenen Größen entsprechen:

Größe	Entsprechung
Klein	$\frac{1}{4} \cdot \text{Zeilenabstand}$
Mittel	$\frac{1}{2} \cdot \text{Zeilenabstand}$
Groß	$1 \cdot \text{Zeilenabstand}$

Der Leerraum *Variabel* ist das vertikale Analogon zu **Einfügen** \triangleright **Besondere Formatierung** \triangleright **Variabler horiz. Abstand**⁶⁸. Die Länge des benutzerdefinierten Leerraums darf auch negativ sein. Befindet sich ein Leerraum am Anfang einer Seite,

⁶⁷Formelstile siehe Kap. 17

⁶⁸siehe Kap. 8.2

erscheint er nicht im Ausdruck. Um in diesem Fall dennoch Leerraum zu erstellen, verwendet man im Menü **Einfügen**▷**Besondere Formatierung**▷**Vertikaler Abstand** die Option **Schützen**.

Eine Anwendung vertikaler Leerräume sind Fußnoten. Wenn mehrere davon auf einer Seite erscheinen, werden diese direkt untereinander geschrieben, was das Lesen erschwert. Um nach einer Fußnote Leerraum zu erstellen, fügt man in der Fußnote als Letztes 1.5 mm vertikalen Leerraum ein. Dies geschieht in diesem Dokument mit den L^AT_EX-Vorspannzeilen

```
\let\myFoot\footnote
\renewcommand{\footnote}[1]{\myFoot{#1\vspace{1.5mm}}}
```

automatisch für jede Fußnote. Der Befehl `\vspace{Länge}` erzeugt dabei benutzerdefinierten Leerraum.

Vertikale Leerräume bzw. deren L^AT_EX-Befehle `\vspace` und `\vfill` werden in Formeln ignoriert.

22.17 Worttrennungen, Zeilen- und Seitenumbrüche

22.17.1 Worttrennungen

Kann L^AT_EX ein langes Wort nicht trennen, ragt es über das Zeilenende hinaus⁶⁹. Um L^AT_EX zu sagen, wo es das Wort trennen darf, fügt man an der betreffenden Stelle über das Menü **Einfügen**▷**Besondere Formatierung** eine **Trennmöglichkeit** ein.

Andererseits kann es vorkommen, dass ein Wort nicht oder nur an einer bestimmten Stelle getrennt werden soll.

Dazu ein Beispiel:

Am Ende der Zeile befindet sich ein Fremdwort, welches L^AT_EX richtig trennt: Déjà-vu-Erlebnis

Soll das Wort nicht am Bindestrich getrennt werden, gibt es zwei Möglichkeiten:

- Man benutzt den Box-Befehl `\mbox`⁷⁰. Da diese Box nur eine Zeile enthalten kann, kann der Boxinhalt nicht umgebrochen werden. Man gibt im T_EX-Modus nach „Déjà“ den Befehl `\mbox{` ein. Vor „Erlebnis“ gibt man im T_EX-Modus `}` ein um die Box zu schließen. Die Zeichen *-vu-* sind damit vor einem Zeilenumbruch geschützt. Damit zwischen „Er“ und „lebnis“ getrennt wird, fügt man eine Trennmöglichkeit ein. Hier das korrigierte Beispiel:

⁶⁹Jede Zeile, die breiter als die im Dokument eingestellte Breite ist, ruft L^AT_EX-Warnungen hervor. Im L^AT_EX-Protokoll, das sich im Menü **Dokument**▷**L^AT_EX-Protokoll** befindet, sind das die Warnungen: „Overfull `\hbox`“

⁷⁰siehe Kap. 9.2

Am Ende der Zeile befindet sich ein Fremdwort, welches L^AT_EX richtig trennt: Déjà-vu-Erlebnis

- Man benutzt den Befehl `\nobreakdash` im T_EX-Modus vor jedem Bindestrich an dem nicht umgebrochen werden soll. Das Ergebnis ist dasselbe wie bei Verwendung von `\mbox`:

Am Ende der Zeile befindet sich ein Fremdwort, welches L^AT_EX richtig trennt: Déjà-vu-Erlebnis

22.17.2 Zeilenumbrüche

Möchte man verhindern, dass ein Ausdruck über zwei Zeilen läuft, kann man vor dem Ausdruck mit **Strg-Enter** einen Zeilenumbruch erzwingen. Dazu ein Beispiel:

In diesem Artikel steht viel über die Planung und Konstruktion der Elbbrücke „Blaues Wunder“ geschrieben.

Erzwungene Zeilenumbrüche dehnen die Zeile vor dem Umbruch nicht über die gesamte Breite. Soll ein Zeilenumbruch in der Nähe des Zeilenendes eingefügt werden, kann man den Befehl `\linebreak` verwenden. Dadurch wird die Zeile immer auf die komplette Breite gedehnt. `\linebreak` wird im T_EX-Modus eingegeben und besitzt folgendes Befehlsschema:

`\linebreak[Dringlichkeit]`

Die Dringlichkeit kann mit einer ganzen Zahl zwischen 0 und 4 angegeben werden. Ist 4 oder nichts angegeben, wird immer umgebrochen. Bei den Dringlichkeitsstufen 0 - 3 entscheidet L^AT_EX, ob umgebrochen wird. Hier das obige Beispiel mit `\linebreak`:

In diesem Artikel steht viel über die Planung und Konstruktion der Elbbrücke „Blaues Wunder“ geschrieben.

Zeilenumbrüche in eingebetteten Formeln sind nur nach Relationszeichen und Binäroperatoren möglich. Das gilt aber nur, wenn sich diese nicht innerhalb anderer Formelkonstrukte wie z. B. in Brüchen oder Klammern befinden. Zur Feineinstellung kann man im L^AT_EX-Vorspann mit den Befehlen

`\relpenalty=Wert` und `\binoppenalty=Wert`

einen Wert für die Dringlichkeit des Zeilenumbruchs nach Relationszeichen bzw. Binäroperatoren festlegen. Der Maximalwert von 10000 bedeutet, dass nie umgebrochen wird. Die Voreinstellungen lauten `\relpenalty=500` und `\binoppenalty=700`.

Möchte man verhindern, dass eine eingebettete Formel umgebrochen wird, setzt man sie in eine L^AT_EX-Klammer.

22.17.3 Seitenumbrüche

Seitenumbrüche können in LyX über das Menü **Einfügen**▷**Besondere Formatierung**▷**Seitenumbruch** erzwungen werden. Es wird dann im Dokument eine neue Seite begonnen und der Inhalt der vorhergehenden Seite bleibt unverändert.

Verwendet man statt dem Menü den Befehl `\pagebreak`, wird der Inhalt der vorhergehenden Seite auf die komplette Seitenlänge gedehnt. Der Befehl ist daher von Vorteil, wenn der Seitenumbruch in der Nähe eines Seitenendes eingefügt werden soll. `\pagebreak` wird im TeX-Modus eingegeben und besitzt dasselbe Schema wie der Befehl `\linebreak` aus dem vorigen Abschnitt.

Seitenumbrüche in Formeln sind eigentlich verboten, trotzdem gibt es für die mehrzeiligen Formelumgebungen **align**, **alignat**, **flalign**, **multline** und **gather** die Möglichkeit Seitenumbrüche zu setzen. Dazu gibt man am Ende der Formelzeile, nach der umgebrochen werden soll, den Befehl `\displaybreak` ein. Der Befehl besitzt dasselbe Schema wie `\pagebreak`.

Möchte man L^AT_EX entscheiden lassen, an welcher Stelle ein Seitenumbruch in einer Formel sinnvoll ist, gibt man im L^AT_EX-Vorspann den Befehl

`\allowdisplaybreaks[Dringlichkeit]`

ein. Die Dringlichkeit ist eine Zahl im Bereich 1–4. Die 1 bedeutet: „Erlaube Seitenumbrüche, aber vermeide sie so weit wie möglich“. Ist keine Dringlichkeit oder eine 4 angegeben, werden Seitenumbrüche durch L^AT_EX nicht vermeiden. Wenn `\allowdisplaybreaks` verwendet wird, werden auch Formeln in der **eqnarray-Umgebung** umgebrochen.

In L^AT_EX es möglich einen Seitenumbruch nach einer Zeile zu verhindern, indem man nach dem Zeilenumbruchbefehl ein `*` eingibt. Dies wird von LyX aber noch nicht unterstützt.

22.18 Links in DVI- und PDF-Dokumenten

22.18.1 Verlinkte Querverweise

Um in DVI- oder PDF-Dokumenten Links zu erstellen, verwendet man das L^AT_EX-Paket **hyperref**. Dieses erzeugt PDF-Lesezeichen und verlinkt das Inhaltsverzeichnis und alle Querverweise auf Kapitel, Formeln und Fußnoten. Das bedeutet, dass man beim Klick auf die Nummer des Querverweises „siehe Kapitel 1“ in das Kapitel 1 gelangt.

Möchte man die Linkfläche vergrößern, so dass das Wort vor der Nummer mit zum Link gehört, verwendet man den von hyperref bereitgestellten Befehl `\autoref`. Da LyX für Querverweise intern den Befehl `\ref` verwendet, muss man diesen umdefinieren, damit `\autoref` benutzt wird. Dazu gibt man im L^AT_EX-Vorspann die Zeile

`\AtBeginDocument{\renewcommand{\ref}[1]{\mbox{\autoref{#1}}}}`

ein. Dabei ist es wichtig, dass die Zeile *nach* dem Laden von `hyperref` eingegeben wird.

`\autoref` schreibt automatisch den Querverweisnamen in der Dokumentsprache vor die Nummer. Die voreingestellten Querverweisnamen sind in der `hyperref`-Dokumentation aufgelistet. So hat z. B. der Verweis auf ein Kapitel den Namen „Abschnitt“. Soll stattdessen der Name „Kap.“ verwendet werden, definiert man den voreingestellten Namen mit folgender \LaTeX -Vorspannzeile um⁷¹:

`\renewcommand{\sectionautorefname}{Kap.}`

Das Ergebnis ist: „siehe Kap. 1“

Soll z. B. bei Verweisen auf Formeln kein Name geschrieben werden, verwendet man die \LaTeX -Vorspannzeilen

`\newlength{\abc}`
`\settowidth{\abc}{\space}`
`\renewcommand{\equationautorefname}{\hspace{-\abc}}`

`\newlength` erstellt eine neue Länge `\abc`, die durch `\settowidth` den Wert der Breite eines Leerzeichens erhält. Dies ist notwendig, da `\autoref` immer ein Leerzeichen vor die Verweisnummer setzt, egal ob davor ein Name steht. Durch den `\renewcommand`-Befehl besteht der Verweisname nur aus einem negativen Leerzeichen.

Man kann die Namen auch im Dokument umdefinieren, damit bei Aufzählungen von Verweisen der Name nicht bei jedem Verweis geschrieben wird. Dazu gibt man die entsprechenden `\renewcommand`-Befehle im \TeX -Modus ein.

Um eigene PDF-Lesezeichen zu erzeugen, verwendet man den Befehl `\pdfbookmark`, der in [6] beschrieben ist.

22.18.2 Links zu externen Objekten

Um Links zu Webseiten, Emailadressen oder anderen Textdokumenten zu erstellen, verwendet man den vom Paket `hyperref` bereitgestellten Befehl `\href`, der folgendes Schema besitzt:

`\href{Adresse}{Beschreibung}`

`\href` wird im \TeX -Modus eingegeben. Im Dokument wird nur die Beschreibung angezeigt, auf die man klicken kann, um zur Adresse zu gelangen. Der Befehl

`\href{http://www.lyx.org}{Webseite von \LaTeX }`

⁷¹Der Befehl `\renewcommand` besitzt dasselbe Schema wie der in Kap. 20.1 beschriebene Befehl `\newcommand`, mit dem Unterschied, dass `\renewcommand` schon vorhandene Befehle umdefiniert.

ergibt Folgendes: [Webseite von LyX](#)

Soll auf eine Emailadresse verwiesen werden, verwendet man den Adressvorspann **mailto:**. Im Titel dieses Dokuments befindet sich z. B. ein Link zu einer Emailadresse, bei dem zusätzlich schon der Emailbetreff eingestellt ist. Dazu wurde der Befehl

```
\href{mailto:uwestoehr@web.de?subject=LyX-Mathed Dokumentation}
{uwestoehr@web.de}
```

verwendet.

Man kann als Adresse auch Dateien angeben. Am Ende von Kap. 19.5 wird z. B. auf ein anderes PDF-Dokument verwiesen. Verweist man auf eine .exe-Datei, kann man sie auch direkt ausführen lassen. Das geschieht mit dem Adressvorspann **run:**. Der Befehl

```
\href{run:Programm.exe}{Programm}
```

erzeugt einen Link, der die Datei *Programm.exe* direkt ausführt.

Soll auf eine bestimmte Stelle in einem anderen Dokument verwiesen werden, gibt man hinter dem Dateinamen die Marke an, an der das andere Dokument geöffnet werden soll. Das Zeichen # fungiert dabei als Trennzeichen zwischen Dateiname und Marke. Marken werden automatisch erzeugt, wenn ein Dokument mit dem Paket **hyperref** erstellt wurde. Die wichtigsten Marken sind in folgenden Tabellen aufgelistet.

Marke	Objekt
Doc-Start	Dokumentbeginn
page	Seite
equation	Formel
Hfootnote	Fußnote
figure	Abbildung
table	Tabelle

Marke	Objekt
part	Teil
section	Kapitel
subsection	Unterkapitel
subsubsection	Unterunterkapitel
cite	Literatureintrag
Item	Aufzählungseintrag

Hinter der Marke folgt ein Punkt und die Zahl der Marke. Bei Literatureinträgen folgt der Literaturschlüssel statt einer Zahl.

Um z. B. auf die zweite Seite der Datei [Formelnummerierung.pdf](#) zu verweisen, verwendet man den Befehl

```
\href{Formelnummerierung.pdf#page.2}{Seite 2}
```

[Seite 2](#)

Um auf Gleichung (iii) zu verweisen, schreibt man

```
\href{Formelnummerierung.pdf#equation.iii}{Gleichung (iii)}
```

[Gleichung \(iii\)](#)

Um eigene Marken zu setzen, verwendet man den Befehl `\hypertarget`, der in [6] beschrieben ist.

Wenn man dieses Dokument als DVI betrachtet, werden die beiden Beispielverweise nicht funktionieren, denn ein DVI-Programm kann nicht mit PDF-Dokumenten umgehen. Es ist also wichtig, dass nur auf gleichartige Dokumente verwiesen wird. Daher noch ein DVI-taugliches Beispiel: [Seite 2 DVI](#)

A Typographische Hinweise

Dieses Kapitel ist eine Zusammenfassung der wichtigsten typographischen Regeln, die teilweise auch im Duden [8] unter **Textverarbeitung** aufgelistet sind.

- Physikalische Einheiten werden *immer* aufrecht gesetzt⁷²: 30 km/h
Zwischen Wert und Einheit befindet sich der kleinste Leerraum, siehe Kap. 8.1.
- Prozent- und Promillezeichen werden wie physikalische Einheiten gesetzt:
1,2‰ Alkohol im Blut
Sie folgen jedoch direkt auf die Zahl, wenn sie in einer Zusammensetzung oder Ableitung stehen: 3%-ige Rendite, 20%-Grenze
- Das Grad-Symbol folgt immer unmittelbar auf die Zahl: 15°, es sei denn es wird in einer Einheit verwendet: 15 °C
- In Zahlen mit mehr als vier Ziffern wird zur Gruppierung vor jeder dritten Ziffer der kleinste Leerraum eingefügt: 18 473 588
- Für Bemaßungen wie z. B. 120×90×40 cm, verwendet man das gelegte Kreuz ×, das über den Befehl `\texttimes`⁷³ verfügbar ist, mit einigen Tastaturdefinitionen aber auch direkt eingegeben werden kann.
- Bei Datumsangaben befindet sich zwischen Tag und Monat der kleinste Leerraum, zwischen Monat und Jahr ein normales Leerzeichen: 23. 6. 1979
- Funktionen, deren Name aus mehreren Buchstaben besteht, werden aufrecht gesetzt⁷⁴, um Verwechslungen mit Variablen zu vermeiden. Denn $\tan\xi$ könnte auch $t \cdot a \cdot n \cdot \xi$ bedeuten.
- Indizes, die aus mehreren Buchstaben bestehen, werden aufrecht gesetzt: E_{kin}
Komponenten von Matrizen werden kursiv gesetzt: \hat{H}_{kl}
- Der Ableitungs-/Integrationsoperator 'd', die Eulersche Zahl 'e' und die Imaginäre Zahl 'i' sollten aufrecht gesetzt werden, um sie nicht mit anderen Variablen zu verwechseln.

⁷²Schriftstile siehe Kap. 11.1

⁷³siehe Kap. 22.12

⁷⁴Funktionen siehe Kap. 15.1

B Synonyme

Einige Zeichen und Symbole können mit mehreren Befehlen erstellt werden. Damit es nicht zu Verwechslungen kommt, wenn in anderen \LaTeX -Dokumentationen andere Befehle angegeben sind, folgt eine Übersicht der Synonymbefehle.

Befehl	äquivalent zu
<code>\ast</code>	<code>*</code>
<code>\choose</code>	<code>\binom</code>
<code>\geq</code>	<code>\ge</code>
<code>\lbrace</code>	<code>{</code>
<code>\lbracket</code>	<code>[</code>
<code>\leftarrow</code>	<code>\gets</code>
<code>\leq</code>	<code>\le</code>
<code>\lor</code>	<code>\vee</code>
<code>\neq</code>	<code>\not=</code>
<code>\slash</code>	<code>/</code>
<code>\vert</code>	<code> </code>

Befehl	äquivalent zu
<code>\backslash</code>	<code>\\</code>
<code>\dasharrow</code>	<code>\dashrightarrow</code>
<code>\land</code>	<code>\wedge</code>
<code>\rbrace</code>	<code>}</code>
<code>\rbracket</code>	<code>]</code>
<code>\rightarrow</code>	<code>\to</code>
<code>\not</code>	<code>\neg</code>
<code>\ne</code>	<code>\not=</code>
<code>\owns</code>	<code>\ni</code>
<code>\square</code>	<code>\Box</code>
<code>\Vert</code>	<code>\\</code>

Literatur

- [1] MITTELBACH, F. ; GOOSSENS, M.: *The \LaTeX Companion*. Addison Wesley, 2004
- [2] [Beschreibung](#) der mathematischen Fähigkeiten von \LaTeX
- [3] \LaTeX Tipps- und Tricks-[Seite](#)
- [4] [Beschreibung](#) von $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$
- [5] [Auflistung](#) aller in \LaTeX verfügbaren Symbole
- [6] [Dokumentation](#) des \LaTeX -Pakets **hyperref**
- [7] [Beschreibung](#) des in Kap. 10.2 vorgestellten Befehls `\mathclap`
- [8] *Duden Band 1*. 22. Auflage, Duden Verlag

Stichwort- und Befehlsverzeichnis

Å, 38

°, 40

×, 66

¢, 66

€, 34

μ, 66

‰, 66

Akzente, 15

für ein Zeichen, 15

für mehrere Zeichen, 17

für Operatoren, 16

im Text, 38

Anführungszeichen, 68

Apostroph, 40

Befehle

A

`\addtocounter`, 51

`\@addtoreset`, 52

`\aligned`, 48

`\alignedat`, 49

`\allowdisplaybreaks`, 74

`\alph`, 53

`\Alph`, 53

`\arabic`, 52, 53

`\arraycolsep`, 10, 43

`\arraystretch`, 10

`\autoref`, 74

B

`\big`, 11

`\bigl` - `\bigr`, 12

`\bigm`, 12

`\binom`, 6

`\binoppenalty`, 73

`\boldmath`, 31

`\boldsymbol`, 33

`\boxed`, 20

C

`\CD`, 58

`\cases`, 6

`\cdots`, 8

`\cfrac`, 5

`\colorbox`, 22

D

`\dbinom`, 6

`\DeclareMathOperator`, 29, 36

`\definecolor`, 24

`\dfrac`, 4

`\displaybreak`, 74

`\displaystyle`, 1

`\dotfill`, 9

`\dots`, 8

E

`\euro`, 35

F

`\fbox`, 20

`\fcolorbox`, 23

`\frac`, 4

`\framebox`, 20

G

`\gathered`, 49

`\genfrac`, 60

H

`\hdotsfor`, 8

`\hfill`, 18

`\hphantom`, 7

`\href`, 40, 75

`\hrulefill`, 9

`\hspace`, 18, 44

I

`\int`, 26

`\intertext`, 49

J

`\jot`, 14, 42

L

`\ldots`, 8

`\left`, 10, 12, 45

`\lefteqn`, 44

`\leftroot`, 6

`\lim`, 37

<code>\linebreak</code> , 73	<code>\smallmatrix</code> , 10
<code>\linewidth</code> , 25	<code>\split</code> , 49
M	<code>\sqrt</code> , 5
<code>\makebox</code> , 22	<code>\stackrel</code> , 36
<code>\mathbin</code> , 30	<code>\subarray</code> , 28
<code>\mathindent</code> , 41	<code>\substack</code> , 28
<code>\mathop</code> , 30	<code>\sum</code> , 26
<code>\mathsurround</code> , 19	T
<code>\mbox</code> , 22	<code>\tag</code> , 52
<code>\middle</code> , 13	<code>\tbinom</code> , 6
<code>\multlinegap</code> , 47	<code>\texorpdfstring</code> , 62
N	<code>\text</code> , 2
<code>\newcommand</code> , 54	<code>\textbackslash</code> , 11
<code>\nobreakdash</code> , 73	<code>\textcent</code> , 66
<code>\not</code> , 6	<code>\textcircled</code> , 39
<code>\numberwithin</code> , 52	<code>\textcolor</code> , 24
O	<code>\textmu</code> , 66
<code>\officialeuro</code> , 35	<code>\textperthousand</code> , 66
<code>\oldstylenums</code> , 39	<code>\texttimes</code> , 66
<code>\overbrace</code> , 13	<code>\textvisiblespace</code> , 3
<code>\overline</code> , 7	<code>\tfrac</code> , 4
<code>\overset</code> , 16, 17	U
P	<code>\unboldmath</code> , 31
<code>\pagebreak</code> , 74	<code>\underbrace</code> , 13
<code>\parbox</code> , 25	<code>\underline</code> , 7
<code>\phantom</code> , 7	<code>\underset</code> , 16, 17
<code>\prod</code> , 26	<code>\uproot</code> , 6
R	V
<code>\raisebox</code> , 22	<code>\vphantom</code> , 7, 45
<code>\reflectbox</code> , 70	<code>\vspace</code> , 72
<code>\relpenalty</code> , 73	X
<code>\renewcommand</code> , 10, 53, 75	<code>\xleftarrow</code> , 14
<code>\resizebox</code> , 70	<code>\xrightarrow</code> , 14
<code>\right</code> , 10, 12, 45	Begrenzungszeichen , 11
<code>\roman</code> , 53	benutzerdefinierte Befehle , 53
<code>\Roman</code> , 53	mit <code>\newcommand</code> , 54
<code>\root</code> , 5	mit Mathe-Makros, 55
<code>\rotatebox</code> , 69	Binomialkoeffizienten , 6
S	Boxen , 20
<code>\scalebox</code> , 70	Absatz-, 24
<code>\setlength</code> , 19	farbige, 22
<code>\shoveleft</code> , 47	mit Rahmen, 20
<code>\shoveright</code> , 47	ohne Rahmen, 22
<code>\sideset</code> , 16	

- rotierte, 69
 - skalierte, 70
 - Text-, 69
- Brüche, 4
 - selbstdefinierte, 60
- chemische Zeichen
 - Isotope, 7
 - Symbole, 59
- Diagramme, 57
- Exponenten, 4
- Fallunterscheidungen, 6
- Formel
 - abgesetzte, 1
 - durchgestrichene, 61
 - eingebettete, 1
 - fett gedruckte, 31
 - im mehrspaltigen Text, 63
 - in Überschriften, 62
 - lange, 43
 - mehrzeilige, 41
 - align-Umgebung, 46
 - alignat-Umgebung, 46
 - Eqnarray-Umgebung, 41
 - flalign-Umgebung, 46
 - Formelteile, 48
 - Gather-Umgebung, 47
 - Multline-Umgebung, 47
 - Spaltenabstand, 43
 - Text, 49
 - Zeilenabstand, 42
 - mit Variablenbeschreibung, 64
 - rotierte, 71
 - skalierte, 71
 - unterstrichene, 7
- Formelnummerierung, 49
 - benutzerdefinierte, 51
 - eigene Klammern, 51
 - mit Buchstaben, 53
 - mit römischen Zahlen, 53
- Formelstile, 40
- Formeluntergliederung, 50
- Fortsetzungspunkte, 8
- Funktionen
 - Modulo-, 38
 - selbstdefinierte, 36
 - vordefinierte, 36
- Grenzwerte, 37
- griechische Buchstaben, 33
 - aufrechte, 65
 - fett gedruckte, 33
 - große, 33
 - kleine, 33
- Hochstellen, siehe Exponenten
- Indizes, 4
- Integrale, 26
- Ionen, siehe chemische Zeichen
- Isotope, siehe chemische Zeichen
- Klammergrößen
 - automatische, 12
 - manuelle, 11
- Klammern, 11
 - für mehrzeilige Ausdrücke, 12
 - horizontale, 13
 - mehrzeilige, 45
 - vertikale, 11
- Komma, 59
- L^AT_EX-Klammern, 2
- L^AT_EX-Vorspann, 2
- Leerraum
 - horizontaler, 17
 - variabel, 18
 - vordefiniert, 17
 - neben eingebetteten Formeln, 19
 - vertikaler, 71
- Linien, 7
- Links in Dokumenten, 74
- Makros, 55
- Mathematischer Textmodus, 2
- Matrizen, 9
- Minuskelziffern, 39

Operatoren, 26
 binäre, 29
 Grenzen, 27
 große, 26
 selbstdefinierte, 29

Pakete
 aeguill, 68
 amscd, 57
 braket, 60
 calc, 25, 42
 cancel, 61, 66
 color, 22
 eurosym, 34
 graphicx, 69
 hyperref, 40, 62, 74, 79
 icomma, 59
 multicol, 63
 soul, 67
 textcomp, 66
 ulem, 68
 upgreek, 65
 was, 59, 65

Pfeile, 14
 beschriftbare, 14
 diagonale, 15
 horizontale, 14
 vertikale, 15

Platzhalter, 7

Querverweise
 auf Formeln, 50
 verlinkte, 74

Rahmen, siehe Boxen
Relationen, 35

Schriften, 30
 Schriftgrößen, 31
 Schriftstile, 30

Seitenumbrüche, 74
 in Formeln, 74

Sonderzeichen, 38
 sonstige, 40

Summen, 26
Symbole, 34
 aus dem textcomp-Paket, 66
 chemische, 59
 EURO-Symbol, 34
 mathematische, 34
 sonstige, 34

Synonyme, 79

T_EX-Modus, 2

Text
 dopelt unterstrichener, 68
 durchgestrichener, 67
 farbig unterlegter, 67
 farbiger, 24
 gesperrter, 67
 in Formeln, 2, 49, 65
 rotierter, 69
 skalierter, 70
 umrandeter, 21
 unterstrichener, 67

Tiefstellen, siehe Indizes

Tilde, 39, 40

Tipps, 59

Typographische Hinweise, 78

Umlaute, 15

Vektoren, 15
 physikalische, 60

Vergleiche, siehe Relationen

Verneinungen, 6

Worttrennung, 72

Wurzeln, 5

Zahlen
 negative, 59

Zeilenumbrüche, 73
 in Formeln, 73

Zitate, siehe Anführungszeichen