

Einführung in das Textsatzsystem



L^AT_EX



05 – Mathesatz I THE LION'S STRONGEST PAW

22. November 2013

Inhalt

- 1 Inline vs. Display
- 2 Größe von Formeln
- 3 Grundbefehle
- 4 Vektoren, Matrizen, Tensoren
- 5 Setzen physikalischer Einheiten
- 6 Integral-/Differentialrechnung
- 7 Symbole
- 8 Schriften
- 9 SI-Einheiten
- 10 Platz
- 11 Feynman-Graphen
- 12 breqn
- 13 cases
- 14 esvect
- 15 mathtools
- 16 ulem, soul

Motto: L^AT_EX 2012

*–
Zuckerbrot, Peitsche und
Mathesatz.*

– P. Gesang

Inlinemode

- Formeln, die direkt im Fließtext vorkommen
- zu berücksichtigen:
- Formeln sollten kurz sein
- Brüche und Wurzeln vermeiden
- Grenzen werden *neben* Integrale, Summen, Produkte gesetzt
- Alles vermeiden, was die Zeilenhöhe stört (Vektoren, Indizes ...)

Inlinemode

$E = mc^2$ kennt jedes Kind, aber kaum jemand kann wirklich mehr damit anfangen als mit $\int_{-\infty}^{\infty} \sum_{n=1}^5 dx$, wobei diese Formel nun mal gar keinen Sinn ergibt, aber zeigt, wie Grenzen im Mathesatz aussehen.

$E = mc^2$ kennt jedes Kind, aber kaum jemand kann wirklich mehr damit anfangen als mit $\int_{-\infty}^{\infty} \sum_{n=1}^5 dx$, wobei diese Formel nun mal gar keinen Sinn ergibt, aber zeigt, wie Grenzen im Mathesatz aussehen.

$E = mc^2$ kennt jedes Kind, aber kaum jemand kann wirklich mehr damit anfangen als mit

$$\int_{-\infty}^{\infty} \sum_{n=1}^5 dx,$$

wobei diese zweite Formel nun mal gar keinen Sinn ergibt, aber zeigt, wie Grenzen im T_EX-Mathesatz aussehen.

Inlinemode

Der Inlinemode ist über drei Wege zu erreichen:

- `\(Formel\)`
- `\begin{math} formel \end{math}`
- `$Formel$`

Inlinemode

Der Inlinemode ist über drei Wege zu erreichen:

- `\(Formel\)`
- `\begin{math} formel \end{math}`
funktioniert nicht in `alltt`-Umgebung
- `$Formel$`
funktioniert nicht in `alltt`-Umgebung

Inlinemode

Der Inlinemode ist über drei Wege zu erreichen:

- `\(Formel\)`
nicht robust!
- `\begin{math} formel \end{math}`
funktioniert nicht in `alltt`-Umgebung
nicht robust!
- `$Formel$`
funktioniert nicht in `alltt`-Umgebung

robust vs. fragile

- fragile Befehle können Probleme bereiten bei:
 - Schreiben in Datei, Ausgabe in log / Terminal, in Verzeichnissen
 - robuste Befehle sind speziell geschützt
- ⇒ \$ \$ ist meist beste Variante

Umbruch

- Formeln können (im Inline-Modus) von T_EX umbrochen werden an:
- Relationen $=$ $<$ $>$ etc.
- binären Operatoren $+$ $-$ etc.
- zum Vermeiden: Gruppieren mittels $\{ \}$ oder `\mbox{ }`
- beamer-Klasse kann Probleme bereiten (gar kein Umbruch)

Umbruch

- Formeln können (im Inline-Modus) von T_EX umbrochen werden an:
- Relationen = < > etc.
- binären Operatoren + - etc.
- zum Vermeiden: Gruppieren mittels {} oder \mbox{}
- beamer-Klasse kann Probleme bereiten (gar kein Umbruch)

```

Etwas Text bis Zeilenende
$a + b + c$\
Etwas Text bis Zeilenende
${a + b + c}$\
Eine viel zu lange Formel:
${a+b+c+d+e+f+g+h+i+j+k+l+m}
$

```

```

Etwas Text bis Zeilenende  $a + b + c$ 
Etwas Text bis Zeilenende  $a + b + c$ 
Eine viel zu lange Formel:
 $a + b + c + d + e + f + g + h + i + j -$ 

```

Display-Formeln

- Auszeichnung wichtiger Formeln
- Darstellung langer Rechnungen
- komplexe Formeln
- mehrfach indizierte Größen
- geschachtelte Brüche

⇒ Alles, was zu groß oder wichtig für Inline-Modus ist

Display-Formeln

Display-Formeln sind über drei Wege in L^AT_EX zu erreichen:

- `\begin{displaymath} Formel \end{displaymath}`
– abgesetzte Formel ohne Nummerierung
- `\[Formel \]`
– Abkürzung für `displaymath`
- `\begin{equation(*)} Formel \end{equation(*)}`
abgesetzte Formel mit (ohne) Nummerierung
- `$$ Formel $$`

Display-Formeln

Display-Formeln sind über drei Wege zu erreichen:

- `\begin{displaymath} Formel \end{displaymath}`
– abgesetzte Formel ohne Nummerierung
- `\[Formel \]`
– Abkürzung für `displaymath`
- `\begin{equation} Formel \end{equation}`
abgesetzte Formel mit Nummerierung
- ~~`$$ Formel $$`~~ **T_EX-Syntax!**
in L^AT_EX führt Verwenden von `$$` `$$` zu unerwarteten und unerwünschten Ergebnissen; unbedingt vermeiden!

Option fleqn

- oft sehen Formeln zentriert nicht gut aus
 - „zerfledderter“ Eindruck
 - linksbündige Ausrichtung oft besser
- ⇒ `fleqn` als Dokumentoption (`\documentclass[fleqn]{scrartcl}`)
- ! funktioniert *nicht* mit `$$` `$$`
 - Paket `nccmath` bietet Umgebung `fleqn` für lokalen Einsatz

Display in Inline und umgekehrt

Große Formeln im Inline-Modus:

- `\displaystyle`

Eine Zeile Text vor dem großen Bruch, damit deutlich sichtbar wird, warum man so große Brüche: `\frac{a}{b} <`
`\displaystyle{\frac{a}{b}}` nicht im Fließtext setzt. Und danach und noch ein bisschen Text für die zweite Zeile, die deutlich mehr von der ersten getrennt ist als die dritte von der zweiten.

Eine Zeile Text vor dem großen Bruch, damit deutlich sichtbar wird, warum man so große Brüche: $\frac{a}{b} < \frac{a}{b}$ nicht im Fließtext setzt. Und danach und noch ein bisschen Text für die zweite Zeile, die deutlich mehr von der ersten getrennt ist als die dritte von der zweiten.

Display in Inline und umgekehrt

Eher nützlich: Kleine Formeln im abgesetzten Modus:

- `\textstyle`

```
\[\frac{1}{2} a > \frac{1}{2} b > \frac{1}{2} c \ \ \text{vs.} \ \ \frac{1}{2} a > \frac{1}{2} b > \frac{1}{2} c\]
```

$$\frac{1}{2}a > \frac{1}{2}b > \frac{1}{2}c \quad \text{vs.} \quad \frac{1}{2}a > \frac{1}{2}b > \frac{1}{2}c$$

Display in Inline und umgekehrt

Eher nützlich: Kleine Formeln im abgesetzten Modus:

- `\textstyle`

```
[\frac 12 a > \frac 12 b > \frac 12 c \ \ \text{~~vs.~~} \
\{ \textstyle \frac 12 a > \frac 12 b > \frac 12 c }\]
```

$$\frac{1}{2}a > \frac{1}{2}b > \frac{1}{2}c \quad \text{vs.} \quad \frac{1}{2}a > \frac{1}{2}b > \frac{1}{2}c$$

Nützliche Definition z. B.

```
\newcommand\half{\textstyle{\frac 1 2}}
```

- `amsmath` bietet `\tfrac` und `\dfrac`

Mehrzeilige Formeln

- Reihe von untereinander angeordneten, zueinander ausgerichteten Gleichungen, verwendet für:
- Herleitungen
- Übersicht
- Vergleich von Formeln

eqnarray

- eqnarray: (veraltete) Standardumgebung für mehrzeilige Formeln
 - führt zu inkonsistenten Abständen!
- ⇒ Artikel in der DTK: „**Vermeidet eqnarray**“
- ⇒ besser: align aus dem amsmath-Paket:

```
\begin{align}
a &= b, \\
c &= d, \\
abc &= d \\
&= r
\end{align}
```

$$\begin{aligned} a &= b, & (1) \\ c &= d, & (2) \\ abc &= d & (3) \\ &= r & (4) \end{aligned}$$

{align*}: keine Nummerierung

einzelne Nummerierung anpassbar mittels \tag{} bzw. \notag

amsmath

- Paket von der American Mathematical Society (AMS)
- besteht aus mehreren Teilpaketen, u. a.:
amsmath, amssymb, amsfonts
- bietet umfangreiche Erweiterungen des Mathesatzes:
- vielfältige Umgebungen und Anpassungen
- neue oder verbesserte Definitionen von Befehlen
- Korrekturen von Abständen
- ...

Abstände

- Abstände werden von T_EX bzw. L^AT_EX bzw. geladenen Paketen kontrolliert
- zu berücksichtigen: verschiedene Abstände bei Variablen, Operatoren, Relationen etc.
- festgelegt durch die `\mathcodes` der Zeichen
- änderbar mit `\kern`, `\,`, `\,`, `\,`; etc.
- **niemals** Konstrukte wie `\ \ \ \` verwenden!
- Besser: `\quad`, `\qquad`, `\hspace{1em}`

Größenänderungen

- Standardbefehle wie `\small`, `\tiny`, `\Huge` haben *innerhalb* von Formeln keine Wirkung
- *Aber*: Formeln passen sich der Umgebungsschrift an:

```
\small\[E = \Huge mc^2\  
\Huge\[E = mc^2\  

```

$$E = mc^2$$

$$E = mc^2$$

- Paket `relsize` bietet `\mathlarger`
- Paket `exscale` bietet definiert größere Versionen von Operatoren (meist nicht nötig)

Punkt vs. Komma

im amerikanischen Satz:

\$1 , 234 . 567 \$

1, 234.567

Punkt vs. Komma

im amerikanischen Satz:

\$1 , 234 . 567 \$

1, 234.567

im deutschen Satz:

\$1 . 234 , 567 \$

1.234, 567

⇒ falsche Spationierung!

Punkt vs. Komma

- Anpassung manuell oder auf Paketebene möglich
 - besser: logische Eingabe mittels Markup
 - genaue Ausgabeformatierung internen Befehlen überlassen
- ⇒ Trennung von Eingabe und Ausgabe
- ⇒ konsequentes Layout
- ⇒ spätere Anpassung für anderssprachigen Satz möglich
- ⇒ `\num` aus dem `siunitx`-Paket (s. u.)

Hoch- und Tiefstellung

- Zeichen mit besonderer Bedeutung: $\hat{}$ und $_$
- Hochstellung: a^b , Tiefstellung: a_b : a_b
- mit `unicode-math` ist unicode-Eingabe einzelner Zeichen möglich:
 $^2 = \hat{2}$, $^{2\,3} \approx \hat{\{23\}}$
- Gruppierungen sind möglich: $a^{\{bc\}}$, $a_{\{bc\}}$: a_{bc}
- Kombination ist möglich: a^b_c : a_c^b
- Ohne vorhergehendes Zeichen: $\hat{\{235\}}U$: ^{235}U
 $\${\vphantom{\frac{1}{2}}}\hat{2}\ \frac{1}{2}$
- Schachtelung nur mit Gruppierung:

$$a_{\{b_{\{c_{\{d_{\{e_{\{f^g\}}\}}\}}\}}^{\{h^{\{i^{\{j_k\}}\}}\}} a_{b_{c_{d_{e_{f^g}}}}}^{h^{i^{jk}}}$$

a_b_c produziert Fehler!

Operatornamen

- Operatornamen werden aufrecht gesetzt und sind vordefiniert:
- $\sin(x)$ vs. $\sin(x)$
- `\sin \cos \tan \lim \atan \arctan`
`sin cos tan lim arctan`
- Paket `amsopn` bietet weitere Vordefinitionen:

```
\arccos \arcsin \arg \cos \cot \coth \deg \det  
\exp \gcd \inf \injl \lg \lim \limsup \ln  
\max \min \projlim \sec \sinh \sup \tanh
```

Definieren von Operationen

Sollten die vorgegebenen Definitionen nicht genügen:

```
\usepackage{amsopn}  
\DeclareMathOperator{\Res}{Res}
```

in der Präambel.

links und rechts

- Klammerung von großen Ausdrücken kann Probleme bereiten:

```
\[(\frac{\int^a x dx}{\sum_{n=1} x})\]
```

$$\left(\frac{\int^a x dx}{\sum_{n=1} x}\right)$$

- Besser:

```
\[ \left( \right. \\ \frac{\int^a x dx}{\sum_{n=1} x} \\ \left. \right) \]
```

$$\left(\frac{\int^a x dx}{\sum_{n=1} x} \right)$$

links und rechts

- `\left` und `\right` vor allem, was dehnbar ist
- `\left(\right)` funktioniert auch
- `\left. \right)` liefert angepasste rechte Klammer
- Punkt als Platzhalter, um Anfang oder das Ende zu markieren
- Hoch- und Tiefstellung werden angepasst:

```
\[\left[\int^a\right]\]
\[\left.\int^a\right.\]
\[\left.\int^a dx\right|^5
_1\]
```

$$\left[\int^a\right]$$

$$\left.\int^a\right.$$

$$\left.\int^a dx\right|^5_1$$

Operatoren

- Operatoren sind intuitiv per Namen zugänglich
- Grenzen per `\limits` angeben
- Mehrzeilige Grenzen mit `\atop`

```
\[\int^X \int\limits^X  
\sum_{n=1}^\infty  
\prod_{n = 1 \atop m = 2}\]
```

$$\int^X \int^X \sum_{n=1}^{\infty} \prod_{\substack{n=1 \\ m=2}}$$

Sonderzeichen

- Viele Zeichen sind über ihren Namen erreichbar,
- z. B. griechische Groß- und Kleinbuchstaben:

```
\[\nabla \pm \mp
\alpha \beta \gamma
\rho \varrho \kappa \vartheta
\varkappa
\epsilon \varphi \theta
\vartheta
A B \Gamma]
```

$$\nabla \pm \mp \alpha \beta \gamma \rho \varrho \kappa \vartheta \epsilon \theta \vartheta A B \Gamma$$

Beachte die Varianten für ϵ , φ , ϑ und ω !

Wurzeln

- Wurzel:
- zu tiefe Unterlängen sind unschön \Rightarrow `\smash`

```
\[
\sqrt[3]{a_{n_{m_p}}}\quad
\quad\quad\quad
\quad\quad\quad
\sqrt{\smash[b]{a_{n_{m_p}}}}
\]
```

$$\sqrt[3]{a_{n_{m_p}}} \quad \sqrt{a} \quad \sqrt{a_{n_{m_p}}}$$

Vektoren

Vektoren sind vielfältig darstellbar:

- Mit Pfeil drüber als `\vec`

```
$\vec a\ \pmb a\ \mathbf a$  
$\underbar a$
```

 \vec{a} ***a*** ***a*** a

Matrizen

```
\[ \begin{matrix}
a_{11} & a_{12} \\
a_{21} & a_{22}
\end{matrix} \]
```

$$\begin{matrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{matrix}$$

Matrizen

```
\left(\begin{matrix}  
a_{11} & a_{12}\\  
a_{21} & a_{22}\\  
\end{matrix}\right)
```

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

Matrizen

amsmath definiert weitere Matrixumgebungen:

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

`pmatrix`

$$\left\| \begin{matrix} a & b \\ c & d \end{matrix} \right\|$$

`Vmatrix`

$$\left| \begin{matrix} a & b \\ c & d \end{matrix} \right|$$

`vmatrix`

$$\left\{ \begin{matrix} a & b \\ c & d \end{matrix} \right\}$$

`Bmatrix`

$$\left[\begin{matrix} a & b \\ c & d \end{matrix} \right]$$

`bmatrix`

$$\begin{matrix} a & b \\ c & d \end{matrix}$$

`smallmatrix`

Satz komplexer Matrizen

```

\[\begin{pmatrix}
a & b & \dots & z\\
b & \dots & \dots & z\\
\vdots & \ddots & \vdots & \vdots\\
\vdots & \vdots & \vdots & \vdots\\
z & b & \dots & \begin{pmatrix} a & b \\ c & d \end{pmatrix}
\end{pmatrix}
\]

```

$$\begin{pmatrix} a & b & \dots & z \\ b & \dots & \dots & z \\ \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ z & b & \dots & \begin{pmatrix} a & b \\ c & d \end{pmatrix} \end{pmatrix}$$

SI-Einheiten

- Setzen von physikalischen Größen erfordert Aufmerksamkeit:
- richtige Abstände, konsistente Schreibweise, korrekte Schrift, ...
- z. B.:

51.3\pm 0.2\,, \text{m}/\text{s}

51.3\pm 0.2\,, \text{m}\text{s}^{-1}

51.3(2)\,, \text{m}\text{s}^{-1}

51.3(2)\,, \frac{\text{m}}{\text{s}}

SI-Einheiten

- Setzen von physikalischen Größen erfordert Aufmerksamkeit:
- richtige Abstände, konsistente Schreibweise, korrekte Schrift, ...
- z. B.:

51.3\pm0.2\,,\,\text{m}/\text{s}

51.3\pm0.2\,,\,\text{m}\text{s}^{-1}

51.3(2)\,,\,\text{m}\text{s}^{-1}

51.3(2)\,,\,\frac{\text{m}}{\text{s}}

⇒ Paket siunitx nimmt fast alle Arbeit ab:

\SI{51.3(2)}{\meter \per \second}

(bei Verwendung von \per müssen Einheiten ausgeschrieben werden, sonst reicht m s)

Grundbefehle aus siunitx

- `\ang[options]{ angle }`
- `\num[options]{ number }`
- `\si[options]{ unit }`
- `\SI[options]{ number }[pre-unit]{ unit }`
- `\numlist[options]{ numbers }`
- `\numrange[options]{ numbers }{ number2 }`
- `\SIlist[options]{ numbers }{ unit }`
- `\SIrange[options]{ number1 }{ number2 }{ unit }`
- `\sisetup{ options }`
- `\tablenum[options]{ number }`

Anpassung von siunitx

- siunitx bietet (*sehr*) umfangreiche Anpassungsmöglichkeiten
- gutes, strukturiertes Lesen/Suchen in der Dokumentation nötig!
- Beispiel: Abtrennen des Fehlers: $(5.35 \pm 0.03) \text{ m}$ statt $5.35(3) \text{ m}$

Anpassung von siunitx

- siunitx bietet (*sehr*) umfangreiche Anpassungsmöglichkeiten
 - gutes, strukturiertes Lesen/Suchen in der Dokumentation nötig!
 - Beispiel: Abtrennen des Fehlers: $(5.35 \pm 0.03) \text{ m}$ statt $5.35(3) \text{ m}$
- ⇒ Suche nach „error“

Anpassung von siunitx

- siunitx bietet (*sehr*) umfangreiche Anpassungsmöglichkeiten
 - gutes, strukturiertes Lesen/Suchen in der Dokumentation nötig!
 - Beispiel: Abtrennen des Fehlers: (5.35 ± 0.03) m statt 5.35(3) m
- ⇒ Suche nach „error“ ⇒ erfolglos
- ⇒ Suche nach „separate“

Anpassung von siunitx

- siunitx bietet (*sehr*) umfangreiche Anpassungsmöglichkeiten
 - gutes, strukturiertes Lesen/Suchen in der Dokumentation nötig!
 - Beispiel: Abtrennen des Fehlers: $(5.35 \pm 0.03) \text{ m}$ statt $5.35(3) \text{ m}$
- ⇒ Suche nach „error“ ⇒ erfolglos
- ⇒ Suche nach „separate“ ⇒ führt zu `separate-uncertainty` ⇒ ☺

Anpassung von siunitx

- siunitx bietet (*sehr*) umfangreiche Anpassungsmöglichkeiten
 - gutes, strukturiertes Lesen/Suchen in der Dokumentation nötig!
 - Beispiel: Abtrennen des Fehlers: $(5.35 \pm 0.03) \text{ m}$ statt $5.35(3) \text{ m}$
- ⇒ Suche nach „error“ ⇒ erfolglos
- ⇒ Suche nach „separate“ ⇒ führt zu separate-uncertainty ⇒ ☺
- Optionen sind lokal einstellbar:
`\SI[separate-uncertainty]{5.35(3)}{m}`
 - oder global:
`\sisetup{separate-uncertainty,exponent-product=\cdot}`

Integrale

amsmath bietet weitere Integrale:

```
\[ \iint \iiint \iiint \oint
\]
\[ \idotsint \int \int \]
```

$$\iint \iiint \iiint \oint$$

$$\int \cdots \int \int \int$$

Integrale

Zusätzliche
Integraldarstellungen bieten:

- wasysym
- txfonts
- esint
- MnSymbol
- mathdesign

Auf Kompatibilität achten
Verschiedene Matheschriften
zusammen können Probleme bereiten.

Relationen

```
$= \equiv \approx \asymp \backslash  
bowtie \cong \dashv \doteq  
\sim \simeq \propto \smile$
```

$$= \equiv \approx \asymp \bowtie \cong \dashv \doteq \sim \simeq \propto \smile$$

Relationen

```
$= \equiv \approx \asymp \backslash  
bowtie \cong \dashv \doteq  
\sim \simeq \propto \smile$
```

$$= \equiv \approx \asymp \bowtie \cong \dashv \doteq \sim \simeq \propto \smile$$

allgemeine Negierung mit `\not`

```
%\not = \neq \not\equiv  
%\not \approx \not A  
%\not\kern-.2em\int \not\kern-.2em\partial \not \smile$
```



Relationen

```
$= \equiv \approx \asymp \bowtie \cong \dashv \doteq \sim \simeq \propto \smile$
```

$$= \equiv \approx \asymp \bowtie \cong \dashv \doteq \sim \simeq \propto \smile$$

allgemeine Negierung mit `\not`

```
%\not = \neq \not\equiv
%\not \approx \not A
%\not\kern-.2em\int \not\kern-.2em\partial \not \smile$
```



Stapeln von Symbolen

```
$_\stackrel{\text{oben}}{\underset{\text{unten}}{a}}$
$_\stackrel{\text{oben}}{\underset{\text{unten}}{a}} = \ddot{a} \neq \dot{\neq}$
```

$$\begin{array}{l} \text{oben} \\ \text{unten} \end{array} \overset{e}{a} = \ddot{a} \neq \dot{\neq}$$

Satz von bra und ket

erster Ansatz:

$$\$|a\rangle \langle a|A|a\rangle\$$$

$$|a\rangle \langle a|A|a\rangle$$

Satz von bra und ket

erster Ansatz:

$$\langle a | A | a \rangle$$

$$\langle a | A | a \rangle$$

zweiter / dritter Ansatz:

$$\langle a | \frac{A}{B} | a \rangle$$

$$\langle a | \frac{A}{B} | a \rangle$$

Satz von bra und ket

erster Ansatz:

$$\$|a\rangle \langle a|A|a\rangle\$$$

$$|a\rangle \langle a|A|a\rangle$$

zweiter / dritter Ansatz:

$$\$\left|a\right\rangle \left\langle a\right|\frac{A}{B}\left|a\right\rangle \$$$

$$|a\rangle \langle a|\frac{A}{B}|a\rangle$$

Guter Ansatz: Paket braket

$$\backslash\mathrm{bra}\ a\ \backslash\mathrm{ket}\ b\ \backslash\mathrm{Bra}\{\Psi_c\}\ \backslash\mathrm{Ket}\{\Psi_{lmn}\}$$

$$\backslash\mathrm{braket}\{a|\frac{A}{B}|a\}\ \backslash\mathrm{Braket}\{a|\frac{A}{B}|a\}$$

$$\backslash\mathrm{set}\{x\ \mathrm{in}\ \mathbb{R}\}\ \backslash\mathrm{Set}\{x\ \mathrm{in}\ \mathbb{R}^2\ |\ \sqrt{x^3}\ \mathrm{in}\$$

(Versionen mit Großbuchstaben passen die Klammersngrößen an ihr Argument an!)

Akzente

Für Operatoren benötigt man z. B. das „Dach“:

```
$\hat{\mathrm{A}} \ \bar{h} \ \check{a}
a \dot{a}
\ddot{a} \ \dddot{a} \ \ddddot{a}
\underbrace{E = mc^2}_{\text{nach Einstein}}
\overbrace{\int_{-\infty}^{\infty}}^{\text{Hinweis}}
```

 \hat{A}
 \bar{a}

Hinweis

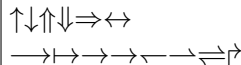
 $z \}$
 $E = mc^2$
 $\int_{-\infty}^{\infty}$

nach Einstein

Pfeile

Für Spinzustände oft verwendete Notation mittels Pfeilen:

```
$\uparrow \downarrow \Uparrow \Downarrow
\rightarrow \leftrightarrow \longrightarrow \mapsto \to \rightsquigarrow
\leftharpoonup \rightharpoonup \rightleftharpoons
\Rsh$
```



mehr Pfeile

Über- und Unterschreibungen von Pfeilen
(Beschriftung von Reaktionsgleichungen etc.)

```
$\xleftarrow[unten]{oben}$  
$\xrightarrow[unten]{}$
```

$$\xleftarrow[unten]{oben}$$

```
$\overleftarrow{a}$  
$\overleftrightharpoonup{b}$  
$\stackrel{\leftrightarrow}{\overleftarrow{a}} \stackrel{\leftrightarrow}{\overrightarrow{b}} T$
```

$$\overleftrightarrow{a} \overleftrightarrow{b} T$$

Unicode-Math

- mittels `unicode-math` können direkt OpenType-Matheschriften verwendet werden:
- `\setmathfont{Latin Modern Math}` (oder z. B. XITS math)
- Vorteile: Zeichen sind an fest definierten Positionen
- Eingabe- und Ausgabekodierung sind eindeutig verknüpft
- falls möglich, Eingabe mit Unicode-Zeichen möglich
- große Anzahl an Zeichen in *einer* Schrift
- viele Schriftschnitte (double struck, italic, bold, etc.) in *einer* Schrift
- Schnitte für wesentlich mehr Zeichen als normal üblich (z. B. aufrechte fette griechische Buchstaben)
- ...

Unicode-Math

Zeicheneingabe mit entsprechender Tastatur direkt möglich:

- $\int \forall \mathbb{C} \Omega \Rightarrow \int \forall \mathbb{C} \Omega$
- Inline- und Displaymodus setzen gleiche Eingabe korrekt um:

$$\int \forall \mathbb{C} \Omega \Rightarrow \int \forall \mathbb{C} \Omega$$

- normale Eingabe mittels Befehlen (wie z. B. von detexify erfahren) weiterhin möglich
- ⇒ erst den Befehl probieren, *ohne* ein Zusatzpaket zu laden!

Unicode-Math

verschieden Schriftschnitte mit „intuitiven“ Namen zugänglich:

```
\mathup, \mathbfup, \mathit, \mathbfit, \mathsfup,  
\mathsfit, \mathbfsfup, ..., \mathbffrak, ...
```

⇒ siehe Dokumentation von unicode-math

Matheschriftpakete

- Matheschrift muss am Anfang des Dokumentes festgesetzt werden
- Kann nicht im Dokument geändert werden
- Pakete freier Schriften (nicht-unicode-basiert!)
- mathpazo
- cmbright
- mathpazo
- eulervm

Eine Reihe nichtfreier Schriften ist in speziellen Paketen verfügbar.

Matheschriften

Hervorhebungen/besondere Buchstaben:

- Kalligraphische Buchstaben `\mathcal`
- Serifenlose
- Fraktur (\mathfrak{R} , \mathfrak{J})
- Aufrechte Buchstaben
- Fettdruck (für Griechisch: Paket `\bm`)
- „blackboard bold“: \mathbb{R}

Setzen von Einheiten

Paket siunitx (Joseph Wright)

```
%%\SI[sepperr]{23.448(5)e23
}{g.cm^3}
%%\si[per=frac]{\joule\per\
eV}
\si{\joule\per\eV}
%\num[dp=2]{4.4583 x 3.2
e21}
\num[mode=text]{4.58}
%\num[expproduct=cdot]{1e10
}
\ang[] {45}
```

$$\text{J eV}^{-1}$$

$$4.58$$

$$45^\circ$$

Ändern der Voreinstellungen mittels `\sisetup`

```
%\sisetup{colourneg}
$\num{-3}, \num{3},
%\num[negcolour=blue]{-5x5}
$,
\num{2}\cdot\num{2}$\\

\def\a{5.1}
$\SI{\a x 5.3}{\milli\metre}
$\\
$\num{\a x 5.3}\si{\square\milli\metre}
$\\
$\num{\a x 5.3}\si{\milli\metre\squared}$
```

$$-3, 3, 2 \cdot 2$$

$$5.1 \text{ mm} \times 5.3 \text{ mm}$$

$$5.1 \times 5.3 \text{ mm}^2$$

$$5.1 \times 5.3 \text{ mm}^2$$

Gradangaben

```
\ang{10}
\ang{12.3}
\ang{4,5}
\\ Heidelberg:
\ang{49;25;}N, \ang{8;43;}O
, \ang{49;25;}N, \ang
{8;43;}O
```

10° 12.3° 4.5°

Heidelberg: 49°25′N, 8°43′O, 49°25′N,
8°43′O

Einheiten

```
\SI{5.54}{ms^{-2}}\\
\SI{5.54}{m s^{-2}}\\
\SI{5.54}{m.s^{-2}}\\
%\SI[valuesep=thick]{5.54}{
m.s^{-2}}\\
%\SI[valuesep=thin]{5.54}{m
.s^{-2}}\\
```

 5.54 ms^{-2}
 5.54 m s^{-2}
 5.54 m s^{-2}

Einheiten

```
%\sisetup{per=fraction}
\SI{1.23}{\joule\per\mole\per\kelvin}

%\ \sisetup{per=slash}
\SI{1.23}{\joule\per\mole\per\kelvin}

%\ \sisetup{per=fraction,
fraction=nice}
\SI{1.23}{\joule\per\mole\per\kelvin}
```

```
1.23 J mol-1 K-1
1.23 J mol-1 K-1
1.23 J mol-1 K-1
```

Änderung der Platzverteilung

- Kerning
- `v/hspace`: `\hspace{1cm}`, `\hspace*{1cm}`
- Achtung bei `\vspace`: Nur im vertikalen Modus möglich
- Phantome

Phantome

```
$a_x = b$\\
$\hphantom{a_x} = b$\\
$\underline{a_x} = \underline{b}$\\
\vphantom{a_x} c \underline{a_x} \\
\underline{b}$
```

$$\begin{aligned}
 a_x &= b \\
 &= b \\
 \underline{a_x} &= \underline{b} \underline{c} \underline{a_x} \underline{b}
 \end{aligned}$$

```
\begin{align*}
a &= b \\
c &= d \\
\int a &= b \\
\end{align*}
```

$$\begin{aligned}
 a &= b \\
 c &= d \\
 \int a &= b
 \end{aligned}$$

Phantome

```
$a_x = b$\
$\hphantom{a_x} = b$\
$\underline{a_x} = \underline{b}\
\vphantom{a_x}}\underline{b}$
```

$$\begin{aligned} a_x &= b \\ &= b \\ \underline{a_x} &= \underline{bb} \end{aligned}$$

```
\begin{align*}
a \&= b\
\vphantom{\int} c \&= d\
\int a \&= b
\end{align*}
```

$$\begin{aligned} a &= b \\ c &= d \\ \int a &= b \end{aligned}$$

Feynman-Graphen

- verschiedene Möglichkeiten für Feynman-Graphen:
- Paket `feynmf`
- Paket `feyn`
- Graphiksoftware
- Metafont
- TikZ/PS-Tricks
- ...

Umbruch von Formeln

- nicht nur Text, sondern auch lange Formeln müssen umbrochen werden
- sinnerhaltender Umbruch schwer
- Umbruch nur im Inline-Mode
- Umbruch nur bei binären Operatoren

Umbruch von Formeln

- `breqn` ermöglicht Umbruch in Display-Formeln
- eigene Umgebungen: `dmath(*)` (wie `\[\]`)
- `dseries`
- `dgroup` (wie `align`)
- `darray` (wie `eqnarray`)
- `dsuspend` (unterbricht)
- Befehl `\condition` für Bedingungen

Probleme

- `breqn` lädt `flexisym`
 - `flexisym` definiert eigene Mathezeichen
- ⇒ Inkompatibilität mit Schriftpaketen
- speziell **inkompatibel** zu `fontspec` (nicht mehr?)

multi-case equations

- **cases** bietet Nummerierung von case-Konstrukten:

```
\begin{numcases}{E = mc^2}
m \neq 0 & Masselose Teilchen\\
m < 0 & Antiteilchen (?)\\
m > 0 & normale Teilchen
\end{numcases}
```

$$E = mc^2 \begin{cases} m \neq 0 & \text{Masselose Teilchen} & (5) \\ m < 0 & \text{Antiteilchen (?)} & (6) \\ m > 0 & \text{normale Teilchen} & (7) \end{cases}$$

Schöne Vektoren

- manchmal hat man spezielle Anforderungen an die Vektorpfeile
- Paket **esvect** bietet Anpassungen der Pfeilform
- korrekter Satz bei Subskripten wird beachtet

```
$\vv a$
```

```
$\vec a$
```

```
$\vv a$
```

 \vec{a} \vec{a} \vec{a}

- Pfeiltyp über Paketoption [a] bis [h] einstellbar
- mögliche Pfeile: siehe Dokumentation

Schöne Vektoren

Subskripte

- Sternversion `\vv*{}{}` sorgt für passende Subskripte:

```
$\vec{ab}_{\Delta}$\\[-2ex]
$\vv{ab}_{\Delta}$\\[-2ex]
$\vv*{ab}{\Delta}$
```

$$\vec{ab}_{\Delta}$$

$$\overrightarrow{ab}_{\Delta}$$

$$\overrightarrow{ab}_{\Delta}$$

mathtools

- Paket `mathtools` bietet:
- Erweiterungen/Ergänzungen/Bugfixes zu `amsmath`
- fine-tuning des Mathesatzes
- Sammlung von Tricks von Michael J. Downes

mathtools

fine-tuning: smashing

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

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

$$X = \sum_{1 \leq i \leq j \leq n}^{a+b+c+d} X_{ij}$$

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

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

```
\[X = \sum_{\mathclap{1 \leq i \leq j \leq n}}^{a+b+c+d} X_{ij}\]
```

```
%\[X = \smashoperator[r]{\sum_{1 \leq i \leq j \leq n}^{a+b+c+d}} X_{ij}\]
```


mathtools

tags

- Standardform der tags ist nicht immer schön: (4)
- Änderung mittels `amsmath`
„[is] not very user friendly (it involves a macro with three @’s in its name)“
- `mathtools`’ Weg:

```
\newtagform{brackets}{[]{} }
\usetagform{brackets}
\begin{equation}E \neq mc^3\end{
equation}
\newtagform{bfbrackets}[\textbf]{[]{} }
\usetagform{bfbrackets}
\begin{equation}E \neq mc^4\end{
equation}
```

$$E \neq mc^3 \quad [8]$$

$$E \neq mc^4 \quad [9]$$

Streichen

durch, unter, quer, ...

- Pakete `ulem` und `soul` bieten verschiedene Hervorhebungsmakros

Streichen

durch, unter, quer, ...

- Pakete `ulem` und `soul` bieten verschiedene Hervorhebungsmakros
- `ulem`: *underline-emphasize*
- `soul`: *space out, underline*

ulem

- Hauptzweck: Ändern von `\emph` zu `\underline`
- falls nicht gewünscht: `\normalem` oder Option `normalem`
- andere Befehle:

```
\uline{test}
\uuline{test}
\uwave{test}
\sout{test}
\xout{test}
\useunder{\uwave}{\bfseries
}{\textbf}
\textbf{test}
```

test test test test ~~test~~ test

soul

```

\so{letterspacing}
\caps{CAPITALS, Small Caps}
\u{underline}
\st{strikeout}
\hl{highlight}
\sethlcolor{blue}
\setulcolor{red}
\setulcolor{green}
\hl{highlight}

```

letterspacing CAPITALS,
 SMALL CAPS underline ~~strikeout~~
 highlight highlight