Experiences typesetting mathematical physics

Dr. Ulrik Vieth

Stuttgart, Germany

EuroTEX 2009

Overview

- Subject of this talk:
  - experiences typesetting mathematical physics
- In this talk:
  - overview of the project (scope, timeline, milestones)
  - setting up TEX math fonts for typesetting physics
  - improving the appearance of math formulas
- Not in this talk:
  - writing a document class (layout, environments)
  - ensuring consistency (spelling, notation, markup)
  - fiddling with page breaks and figure placement
  - managing and organizing a large-scale project
  - maintaining cross-references and preparing an index
  - dealing with technical difficulties (size, runtime, capacity)

Scope of the project

- Lecture notes in theoretical physics:
  - 5 main courses
    - classical mechanics
    - electrodynamics
    - quantum mechanics
    - thermodynamics
    - quantum field theory
  - 3 special courses
    - special and general relativity
    - cosmology
    - elementary particle theory
- Total size: 2500+ pages
  - first published as 2 big combined volumes
  - later republished as individual volumes

Timeline of the project

- Timeline and Milestones:
  - 1989: begin university studies in math and physics
  - 1990: first contact with TEX, but no computer to run it
  - 1991: got involved in typesetting mathematical physics
  - 1991–1998: employed to work on lecture manuscripts
  - 1991: started the project with LATEX 2.09
  - 1994: converted the project to LATEX 2_ε
    wrote custom document class and macro packages
  - 1997: publisher found, plans to publish 2 big volumes
    reorganized project, switched layout and fonts
  - 1999+2004: volumes 1 and 2 published
  - 2006–2009: individual volumes published

Layout details

- Lecture notes edition (1991–1996)
  - full-page layout on A4 paper
  - typeset with Computer Modern fonts at 11 pt
  - loose spacing (non-zero parskip, no parindent)
  - simple environment markup (headings + end marker)
  - each volume processed as individual books
- Publisher edition (1997–2004)
  - book-size layout on smaller paper (17 $\times$ 24 cm)
  - typeset with Times/MathTime fonts at 10 pt
  - compact spacing (no parskip, non-zero parindent)
  - visual environment markup (examples, exercises)
  - small for non-essential material (examples, proofs)
  - several volumes combined into a single book

Styles of mathematical formulas

- There is no one true universal style of math!
- There are many different styles of math:
  - national styles (based on typographic traditions)
    e.g. American, French, Russian
  - publisher styles for journals, books, series
    e.g. AMS, APS, AIP, IOP, Elsevier, Springer
  - standards for fields of sciences
    e.g. physics (IUPAP), physical chemistry (IUPAC)
  - standards by standard organizations
    e.g. ISO 31-11:1992, ISO 80000-2:2009
- BUT: There is only one default style of math:
  - TEX implements one particular style (American math)
  - TEX does not support other styles equally well

Requirements for typesetting physics

- Guidelines for typesetting physics:
  - physical quantities in math italic
  - physical units in upright roman
  - vectors in bold math italic
  - tensors in bold sans serif italic
  - chemical elements in upright roman
  - elementary particles in upright roman
  - math constants (e, i, π) in upright roman
  - math operators (d, ∂, δ, Δ) in upright roman
- applicable to all letters without exceptions (Latin and Greek, uppercase and lowercase)
- applicable to all documents without exceptions (official guidelines vs publisher styles)

Requirements for typesetting physics

- Variations in publisher styles:
  - physical quantities (mostly) in math italic
  - vectors sometimes in bold upright
  - tensors sometimes in bold sans serif upright
  - elementary particles sometimes in italic
  - math constants/operators sometimes neglected or only partially supported (d, e, i, but not ∂, δ, π)
  - not applicable to all letters without exceptions
- Problems with publisher styles:
  - incomplete support of official guidelines
  - markup not portable across different publisher styles
  - inconsistent markup for physical entities
  - inconsistent markup for math constants/operators

What TEX provides and what is missing

- What TEX provides by default:
  - Latin alphabet in italic, switchable to roman or bold
  - Greek uppercase in roman, switchable to italic or bold
  - Greek lowercase in italic, non-switchable to other fonts
- What is missing in TEX:
  - bold math italic exists, but not available as math alphabet
  - bold sans serif italic not available in CM (but exists in LM)
  - Greek lowercase not available in upright roman or bold
  - Greek lowercase not switchable to other fonts
- Limitations in TEX fonts:
  - some fonts not available or only available in one size
  - different font encodings, different range of alphabets: OML (full Latin+Greek) vs OT1 (Latin+Greek subset)

Defining math fonts and alphabets

- How to define additional math fonts?
  - in LaTeX 2.09: not really supported by format
  - in LaTeX 2_ε: supported by NFSS 2 interface
- Define symbol fonts for vectors and tensors:

```
\DeclareSymbolFont{vectors}{OML}{cmm}{b}{it}
\DeclareSymbolFont{tensors}{OT1}{cmss}{bx}{it}
```

- Define font switches for symbol fonts:

```
\DeclareSymbolFontAlphabet{\mathvec} {vectors}
\DeclareSymbolFontAlphabet{\mathtens}{tensors}
```

- Potential problems:
  - silent font substitutions: `cmss/bx/it -> cmss/bx/n`

Redefining math codes of symbols

- Uppercase Greek is upright by default (`operators`)
  Uppercase Greek should be italic by default (`letters`)

```
\DeclareMathSymbol{\Gamma}  {\mathalpha}{letters}{"00}
\DeclareMathSymbol{\Delta}  {\mathalpha}{letters}{"01}
...
\DeclareMathSymbol{\Omega}  {\mathalpha}{letters}{"0A}
```

- Lowercase Greek is non-switchable by default (`\mathord`)
  Lowercase Greek should be made switchable (`\mathalpha`)

```
\DeclareMathSymbol{\alpha}  {\mathalpha}{letters}{"0B}
\DeclareMathSymbol{\beta}   {\mathalpha}{letters}{"0C}
...
\DeclareMathSymbol{\varphi} {\mathalpha}{letters}{"27}
```

Handling of lowercase Greek

- Lowercase Greek should only switch to some fonts (OML)
  Lowercase Greek should not switch to other fonts (OT1)
- Define test for lowercase Greek:

```
\def\@lowgreektest#1{%
  \setbox\@lowgreekbox=\hbox{$%
    \global\@lowgreekfalse
    \ifnum\alpha>#1\else\ifnum\varphi<#1\else
    \global\@lowgreektrue\fi\fi$}}
```

- Define conditional font switches:

```
\def\@lowgreekswitch#1#2#3{\@lowgreektest{#1}%
  \if@lowgreek\def\next{#3}\else\def\next{#2}\fi
  \next{#1}}
```

Defining font switches and markup

- Conditional font switches (default style):
  - particles in upright roman, but lowercase Greek in italic
  - tensors in bold sans serif, but lowercase Greek in bold italic

```
\DeclareRobustCommand{\particle}[1]{%
  \@lowgreekswitch{#1}{\mathrm}{\mathnormal}}
\DeclareRobustCommand{\tens}[1]{%
  \@lowgreekswitch{#1}{\mathtens}{\mathvec}}
```

- Conditional font switches (publisher styles):
  - vectors in bold upright, but lowercase Greek in bold italic

```
\DeclareRobustCommand{\vec}[1]{%
  \@lowgreekswitch{#1}{\mathbf}{\mathvec}}
```

- Unconditional font switches (default style):

```
\let\vec=\mathvec
```

Defining font switches and markup

- Logical markup for entities in physics:

markup	purpose	font	scope
none (default)	physical quantities	`\mathnormal`	Latin and Greek
`\units`	physical units	`\mathrm`	Latin mostly
`\text`	textual material	`\mathrm`	Latin only
`\chem`	chemical elements	`\mathrm`	Latin only
`\particle`	elementary particles	(conditional)	Latin and Greek
`\vec`	vector quantities	`\mathvec`	Latin and Greek
`\tens`	tensor quantities	(conditional)	Latin and Greek

Mathematical constants and operators

- Guidelines for typesetting physics:
  - math constants (e, i, π) should be upright roman
  - math operators (d, ∂, δ, Δ) should be upright roman
- Problems and limitations:
  - lack of suitable fonts for upright Greek (∂, δ, π)
  - need for author awareness for context-specific markup (d, e, i for physical quantities, d, e, i for math)
  - different markup conventions used by publisher styles (`\d`, `\e`, `\i` vs `\dd`, `\ee`, `\ii` vs `\rmd`, `\rme`, `\rmi`)
  - possible conflicts with standard TEX macros
- Possible workarounds: global changes to math codes

```
\DeclareMathSymbol{d}{\mathalpha}{operators}{‘d}
```

Summary and conclusions

- Situation in the mid-1990s:
  - little choices of math fonts (CM, Concrete, Euler, MathTime)
  - limited range of symbols and math alphabets
  - limited support by macro packages for math fonts
- Situation in recent years:
  - more choices of math fonts (tx, px, pazo, fourier, mathdesign)
  - more extensive range of symbols and math alphabets
  - more extensive macro support (`\upGamma`, `\itGamma`)
  - better support for switching options (slanted vs uprightgreek)
- Conclusions:
  - better font and macro support for typesetting physics
  - BUT: still no comprehensive physics package

Summary and conclusions

- Future considerations:
  - Unicode math has encoded all possible math alphabets
  - OpenType math fonts provide reference implementations
  - XƎTEX and LuaTEX have added support for OpenType math
  - BUT: inconvenient to use math alphabets at U+1Dxxx
- Future work needed:
  - concepts of font switches in math have to be reconsidered
  - font switching macros have to be reimplemented differently
  - font switching macros have to be done only once and for all
- Conclusions:
  - logical markup for entities in physics still needed
  - author awareness still needed for context-specific markup

Improving the appearance of math formulas

- Before attempting to improve math formulas:
  - make sure that math formulas are properly coded
  - make sure that proper markup is used (e.g. `\mathop`)
  - make sure that suitable macros are defined and used
  - read or refresh the math chapters of *The TEXbook*
- Some reasons to improve math formulas:
  - fix inconsistencies in alignment of indices
  - fix inconsistencies in size of delimiters
  - prevent delimiters from becoming too big
  - make delimiters bigger for visual emphasis
  - adjust spacing to emphasize logical structure
  - adjust spacing to remove unexpected visual gaps

Improving the appearance of math formulas

- Fixing inconsistent alignment of indices:
  - alignment of subscripts with or without superscripts:

$$x(t) = x_0 + v_0 t \quad \text{vs} \quad x'(t) = x'_0 + v'_0 t$$

  - 1st solution: add phantom superscripts:

$$x(t) = x_0 + v_0 t \quad \text{vs} \quad x'(t) = x'_0 + v'_0 t$$

  - 2nd solution: use staggered indices:

$$x(t) = x_0 + v_0 t \quad \text{vs} \quad x'(t) = x'_0 + v'_0 t$$

  - problem: need for backspacing with staggered indices
- Evaluating the solutions:
  - 1st solution: easier to use, but more height (in displays)
  - 2nd solution: more difficult, but less height (cramped style)

Improving the appearance of math formulas

- Fixing inconsistent size of delimiters:
  - automatic sizes of delimiter with `\left`, `\right`
  - specific sizes with `\big`, `\Big`, `\bigg`, `\Bigg`
- Problems and solutions with automatic sizes:
  - automatic sizes: depends on ascenders + descenders

$$\left(\frac{\cos\alpha}{\alpha}\right)^2 \quad \text{vs} \quad \left(\frac{\sin\beta}{\beta}\right)^2$$

  - specific sizes: using `\bigg` ($= 24\,\text{pt} = 2\,\text{lines}$)

$$\left(\frac{\cos\alpha}{\alpha}\right)^2 \quad \text{vs} \quad \left(\frac{\sin\beta}{\beta}\right)^2$$

Improving the appearance of math formulas

- Preventing delimiters from becoming too big:
  - automatic sizes of delimiter with `\left`, `\right`
  - specific sizes with `\big`, `\Big`, `\bigg`, `\Bigg`
- Problems and solutions with automatic sizes:
  - automatic sizes: depends on enclosed material

$$R = \left(\sum_{i=1}^{N} m_i r_i \right) / M \,, \qquad M = \sum_{i=1}^{N} m_i$$

  - specific sizes: using `\Big` (= 18 pt = 1.5 lines)

$$R = \Big(\sum_{i=1}^{N} m_i r_i \Big) / M \,, \qquad M = \sum_{i=1}^{N} m_i$$

Improving the appearance of math formulas

- Making delimiters bigger to emphasize structure:
  - automatic sizes of delimiter with `\left`, `\right`
  - specific sizes with `\big`, `\Big`, `\bigg`, `\Bigg`
- Problems and solutions with automatic sizes:
  - automatic sizes: nested delimiters just big enough

$$\delta \int_{t_0}^{t_1} F\left(x(t), \dot{x}(t), t\right) = 0$$

  - specific sizes: using `\big` (= 12 pt = 1 line)

$$\delta \int_{t_0}^{t_1} F\big(x(t), \dot{x}(t), t\big) = 0$$

Improving the appearance of math formulas

- Preventing radicals from becoming too big:
  - automatic sizes: depends on enclosed material
  - specific sizes: not available for radicals
- Problems and solutions with automatic sizes:
  - automatic sizes: depends on placement of indices

$$\sqrt{p^2\,c^2 + m_0^2\,c^4} \qquad \frac{1}{\sqrt{p^2\,c^2 + m_0^2\,c^4}}$$

  - specific sizes: using staggered indices:

$$\sqrt{p^2\,c^2 + m_0{}^2\,c^4} \qquad \frac{1}{\sqrt{p^2\,c^2 + m_0{}^2\,c^4}}$$

Improving the spacing of math formulas

- How spacing of math works in TEX:
  - TEX does spacing of math based on class of symbols Ord, Op, Bin, Rel, Open, Close, Punct, etc.
  - TEX does not understand logical structure of formulas
  - TEX does spacing in terms of boxes of glyph metrics
  - TEX does not understand about visual gaps or collisions
  - TEX does not know about exceptions to the rules
- Reasons for manual spacing and backspacing:
  - emphasizing logical structure of formulas
  - preventing visual gaps (backspacing)
  - preventing visual collisions (spacing)
  - preventing or correcting mis-interpretations

Improving the spacing of math formulas

- Examples of manual spacing:
  - before sentence punctuation in displays
  - before differential quantities in integrals
  - after prefix / before suffix terms
  - before exponential terms
  - for invisible multiplication: $n(n-1)$ vs. $n\,(n-1)$
  - following factorial terms: $n!k!$ vs. $n!\,k!$
  - following radical terms: $\sqrt{2}x$ vs. $\sqrt{2}\,x$
- Examples of manual backspacing:
  - when using staggered indices: ${m_0}^2\,c^4$ vs. ${m_0}^2\,c^4$
  - when using slashed fractions: v^2/c^2 vs. v^2/c^2
  - following exponents in differentials: $\mathrm{d}^3\,V$ vs. $\mathrm{d}^3 V$
  - in exponents attached to big delimiters

Improving the spacing of math formulas

- Examples of manual spacing:
  - without spacing

$$\int F(x,y,z)\mathrm{d}^3\,V = \int F(r,\vartheta,\varphi)r^2\mathrm{d}r\sin\vartheta\mathrm{d}\vartheta\mathrm{d}\varphi.$$

  - with spacing

$$\int F(x,y,z)\,\mathrm{d}^3V = \int F(r,\vartheta,\varphi)\,r^2\mathrm{d}r\sin\vartheta\,\mathrm{d}\vartheta\,\mathrm{d}\varphi\,.$$

- Details:
  - space advisable after prefix term $F(\dots)$, unless provided
  - no space needed after exponent in $r^2\mathrm{d}r$, but before term
  - no space needed before mathop in $\sin\vartheta\,\mathrm{d}\vartheta$
  - space needed before differential terms in $\mathrm{d}\vartheta$ and $\mathrm{d}\varphi$
  - space needed before sentence-end punctuation

Improving the spacing of math formulas

- Examples of manual spacing:
  - without backspacing

$$\psi(x,t) = \int A(k)\mathrm{e}^{\mathrm{i}(kx-\omega(k)t)}\mathrm{d}k.$$

  - with backspacing

$$\psi(x,t) = \int A(k)\,\mathrm{e}^{\mathrm{i}\,(kx-\omega(k)t)}\,\mathrm{d}k\,.$$

- Details:
  - space advisable after prefix term $A(k)$, unless provided
  - space advisable before exponential term $\mathrm{e}^{\mathrm{i}\,(kx-\omega(k)t)}$
  - space needed before differential term $\mathrm{d}k$
  - space needed before sentence-end punctuation

Improving the spacing of math formulas

- Examples of manual spacing:
  - without spacing

$$\mathrm{i}\hbar\frac{\partial\psi}{\partial t} = \frac{1}{2m}\left(\frac{\hbar}{\mathrm{i}}\nabla - qA\right)^2\psi.$$

  - with spacing

$$\mathrm{i}\hbar\,\frac{\partial\psi}{\partial t} = \frac{1}{2m}\left(\frac{\hbar}{\mathrm{i}}\nabla - qA\right)^2\psi\,.$$

- Details:
  - space advisable after factor, before fraction
  - backspace advisable in exponent to big delimiters
  - backspace advisable after big delimiter before ψ
  - space needed before sentence-end punctuation

Summary and conclusions

- Suggestions:
  - many improvements to math formulas are possible
  - many examples already discussed in *The TEXbook*
  - it is worth reading math chapters of *The TEXbook*
  - lots of experience needed for good math typesetting
  - lots of examples are helpful to develop understanding
- Caution:
  - it is easy to go overboard on manual adjustments
  - see the advice on kerning in *The METAFONTbook*
  - only use half as much of what you think looks right