Typesetting with TEX / LATEX

Part IV: Basic Mathematics and AMSLATEX

F. C. Langbein

School of Computer Science
Cardiff University

Version 1.0

Overview

- Part I: basic components and essential LATEX
- Part II: formatting and layout
- Part III: figures and tables
- Part IV: basic mathematics and AMSLATEX
- Part V: PDFLATEX and slides
- Part VI: BIBTEX and MakeIndex
- Part VII: useful things...

Mathematics

Basic Mathematics

- TEX/LATEX uses a special mode for typesetting mathematics
- Different versions of formula environments
  - In-line maths for formula in text:

```
\begin{math} ... \end{math}
\( ... \)          $ ... $
```

  - Display maths as separated one-line formula:

```
\begin{displaymath} ... \end{displaymath}
\[ ... \]           $$ ... $$
```

  - Numbered one-line display maths formula:

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


Simple Maths Examples

- In-line maths:

```
Let $x$ be an integer,
s.t.\ \( x = 2n + 1\)
```

Let x be an integer, s.t. $x = 2n + 1$

- Display maths:

```
We have
$$ f(x) = 4x + 1$$
```

We have

$$f(x) = 4x + 1$$

- Numbered display maths:

```
\begin{equation}
g(x) = (x-1) / 4
\end{equation}
```

$$g(x) = (x - 1)/4 \quad (1)$$

Subscripts and Superscripts

- Subscripts are created by:

```
_{sub-script}
```

- Superscripts are created by:

```
^{super-script}
```

- Sub-/Super-scripts can be nested and grouped
- Example:

```
$$a_l = b_k c^k_j +
d_{x^l} +
{f_l}^t$$
```

$$a_l = b^l_k c^k_j + d_{x^l} + f^t_l$$

Fractions and Square Roots

- Fractions are produced using:

```
\frac{numerator}{denominator}
```

- Roots are produced using:

```
\sqrt[n]{formula}
```

- Example:

```
$$\frac{\sqrt{2 +
z^2}}{\sqrt[3]{a} +
5}$$
```

$$\frac{\sqrt{2+z^2}}{\sqrt[3]{a}+5}$$

Greek Letters and Special Functions

- Greek letters are produced by \ followed by letter name:

```
$\alpha, \beta, \Gamma,
\epsilon, \varepsilon,
\tau$
```

$\alpha, \beta, \Gamma, \epsilon, \varepsilon, \tau$

- Special functions can be produced using commands like \log, \sin, \exp

```
$\exp (i\theta) = \cos
\theta + i \sin \theta$
```

$\exp(i\theta) = \cos\theta + i\sin\theta$

- Two versions of modulus function

```
$a \bmod b$,
$i \pmod{j}$
```

$a \bmod b$, $i \pmod{j}$

Summations, Products, Limits

- Summations and products:

```
\sum_{low}^{high}
\prod_{low}^{high}
```

- Limits:

```
\lim_{limit}
```

```
$$
\lim_{\theta\rightarrow
\pi}
\sum_{i=1}^{n}
\theta^i\sin \theta
$$
```

$$\lim_{\theta\rightarrow\pi}\sum_{i=1}^{n}\theta^i\sin\theta$$

More Summation Like Symbols

- Commands to produce variable size summation like symbols:

\int	integral	\oint	circular integral
\bigcup	big union	\bigcap	big intersec.
\coprod	coproduct ...		

- Note difference between inline and display style:

```
$\lim_{b\to
\infty}\int_a^b f(x)$
$$\lim_{b\to
\infty}\int_a^b f(x)$$
```

$\lim_{b\to\infty}\int_a^b f(x)$

$$\lim_{b\to\infty}\int_a^b f(x)$$

Delimiters

- Brackets around a tall object in math mode does not look right with normal sized brackets:

```
$$(\frac{1}{1 + x})$$
```

$$(\frac{1}{1+x})$$

- Under such circumstances use the commands:

```
\leftDelimiter        \rightDelimiter
```

```
$$\left(\frac{1}{1 +
x}\right)$$
```

$$\left(\frac{1}{1+x}\right)$$

  - Size is adjusted automatically

- left/right pairs have to match (delimiters may be different)

Arrays

- To typeset arrays use array environment
  - Elements are arranged in rows and columns for vectors, matrices, different cases, ...
  - Similar to tabular environment, but inside maths mode
- A plain array:

```
$$\begin{array}{cc}
0 & 1\\
2 & 3
\end{array}$$
```

$$\begin{array}{cc} 0 & 1 \\ 2 & 3 \end{array}$$

Matrices and Vectors

- Use delimiters to get brackets, etc.

```
$$\left(
\begin{array}{cc}
0 & 1\\
2 & 3
\end{array}
\right)\left[ \begin{array}{cc}
4 & 5\\
6 & 7
\end{array} \right\}$$
```

$$\left(\begin{array}{cc} 0 & 1 \\ 2 & 3 \end{array} \right) \left[\begin{array}{cc} 4 & 5 \\ 6 & 7 \end{array} \right\}$$

Invisible Delimiters

- Use `\right.` or `\left.` for an invisible delimiter

```
$$
f(x) = \left\{
\begin{array}{cl}
0 & x \leq 0 \\
1 & x > 0
\end{array}
\right.
$$
```

$$f(x) = \left\{ \begin{array}{cl} 0 & x \leq 0 \\ 1 & x > 0 \end{array} \right.$$

- Do not use `array` for multi-line formulæ

Multiline Formulæ

- Use the `eqnarray` environment for multiple aligned equations
  - Works similar to `array` with three fixed columns: rcl
  - `eqnarray` numbers each line

```
\begin{eqnarray}
\ln (f(x)) & = & x^2 + \frac{1}{x + 3}\\
f(x) & = & \exp \left (x^2 + \frac{1}{x + 3}\right)
\end{eqnarray}
```

$$\begin{eqnarray} \ln(f(x)) & = & x^2 + \frac{1}{x+3} \quad (2) \\ f(x) & = & \exp\left(x^2 + \frac{1}{x+3}\right) \quad (3) \end{eqnarray}$$

Unnumbered Multiline Formulæ

- Use `\nonumber` to suppress line numbering in `eqnarray` for a single line

```
\begin{eqnarray}
\ln (f(x)) & = & x^2 + \frac{1}{x + 3} \nonumber\\
f(x) & = & \exp \left (x^2 + \frac{1}{x + 3}\right)
\end{eqnarray}
```

$$\begin{eqnarray} \ln(f(x)) & = & x^2 + \frac{1}{x+3} \\ f(x) & = & \exp\left(x^2 + \frac{1}{x+3}\right) \quad (4) \end{eqnarray}$$

- Use `\eqnarray*` environment for unnumbered multi-line formula

Text and Stacking

- Include text in formula (also see AMSLATEX):

```
\mbox{some text}
```

- To stack things:

```
\stackrel{top}{bottom}
```

```
H$_2$CO$_3$
$\stackrel{\mbox{\small
heat}}
{\longrightarrow}$
H$_2$0 $+$ CO$_2$
```

$H_2CO_3 \xrightarrow{\text{heat}} H_20 + CO_2$

Symbols, etc.

- Many additional symbols available in maths mode

`\approx`	$\approx$	`\neq`	$\neq$
`\leq`	$\leq$	`\geq`	$\geq$
`\partial`	∂	`\pm`	$\pm$
`\cdots`	$\cdots$	`\vdots`	$\vdots$
`\ddots`	$\ddots$	`\leftarrow`	$\leftarrow$
`\Leftarrow`	$\Leftarrow$	`\longleftarrow`	$\longleftarrow$
`\Longleftarrow`	$\Longleftarrow$	`\rightarrow`	$\rightarrow$
`\mapsto`	$\mapsto$	`\aleph`	$\aleph$
`\forall`	$\forall$	`\exists`	$\exists$
`\cup`	$\cup$	`\cap`	$\cap$
`\setminus`	$\setminus$	`\times`	$\times$

see literature...

AMSLaTeX Extensions

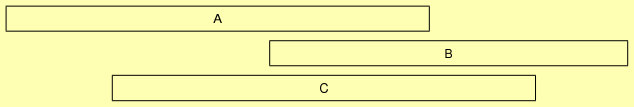
AMSLaTeX Math Extensions

- AMSLaTeX provides various enhancements for improving the information structure and typesetting of LaTeX documents containing advanced mathematics
  - Additional document classes (`amsart`, `amsbook`, ...)
  - Symbol and math fonts (`amsfonts`, `amssymb` packages)
  - Additional commands for typesetting formulas (`amsmath` package)
- We concentrate on the math commands extensions provided by `amsmath` package
- For more information see http://www.ams.org/tex/

Equations without Alignment

- Use `multline` environment to split equations into multiple lines with single equation number
  - `\shoveleft`, `\shoveright` force line to the left or right

```
\begin{multline}
\framebox[.65\columnwidth]{A}\\
\shoveright{\framebox[.55\columnwidth]{B}}\\
\framebox[.65\columnwidth]{C}
\end{multline}
```



(5)

Equations with Alignment

- `split` environment is like `multline`, but allows `&` to mark alignment points

```
\begin{equation*}\begin{split}
f(x) & = \sum_{l = 1, \dotsc, n} {x_l}^l\\
& \quad + \prod_{l = 1, \dotsc, n} x_l\\
\end{split}\end{equation*}
```

$$f(x) = \sum_{l=1,\dots,n} {x_l}^l + \prod_{l=1,\dots,n} x_l$$

Equation Groups without Alignment

- `gather` environment to group consecutive equations without alignment

```
\begin{gather}
  f (x) = \exp (ix) + i\\
  \begin{split}
    g(x) = & \sin(x) + \\& i \cos(x)\\
  \end{split}\\
  h (x) = 1 + 2 + 3 + 4 + 5 + 6 + \cdots + x
\end{gather}
```

$$f(x) = \exp(ix) + i \tag{6}$$

$$g(x) = \sin(x) + i\cos(x) \tag{7}$$

$$h(x) = 1 + 2 + 3 + 4 + 5 + 6 + \cdots + x \tag{8}$$

Equation Groups with Alignment

- `align` environment is used to align multiple equations

```
\begin{align}
x&=y  &  X&=Y  &  a&=b + c\\
x'&=y'  &  X'&=Y'  &  a'+c&=b'
\end{align}
```

$$x = y \qquad X = Y \qquad a = b + c \tag{9}$$

$$x' = y' \qquad X' = Y' \qquad a' + c = b' \tag{10}$$

- `alignat` environment allows to specify horizontal space between equations explicitly:

```
\begin{alginat}{space} ...\end{alignat}
```

Alignment Building Blocks

- `gathered`, `aligned`, `alignedat` to get alignment environment inside an equation, e.g.:

```
\begin{equation}\left.\begin{aligned}
B' &=-\partial \times E,\\
E' &=\partial\times B - 4 \pi j,
\end{aligned}\right\}
\qquad \text{Maxwell's equations}
\end{equation}
```

$$\left.\begin{aligned} B' &= -\partial \times E, \\ E' &= \partial \times B - 4\pi j, \end{aligned}\right\} \quad \text{Maxwell's equations} \qquad (11)$$

- Note, `\text{string}` is used to insert text in formula

Cases and Text

- Case distinction with `\text` and `\intertext`:

```
\begin{gather}$$P_{r-j} = \begin{cases}
0 & \text{if $r - j$ is odd},\\
1 & \text{otherwise,}
\end{cases},\\
\intertext{and}
Q_l = l!\,.
\end{gather}
```

$$P_{r-j} = \begin{cases} 0 & \text{if } r-j \text{ is odd}, \\ 1 & \text{otherwise}, \end{cases} \qquad (12)$$

and

$$Q_l = l!\,. \qquad (13)$$

Matrices

- Matrix environments with build in delimiters:

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

`Bmatrix` $\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$ `vmatrix` $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$

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

  - Like `align`, but more economical spacing
  - No column specification required (max. 10 centred cols)
- `smallmatrix` environment for fitting matrices $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ in text
  - Delimiters have to be added using `\left(`, `\right)`, ...

Math Spacing

- Commands to adjust spacing between symbols in a formula:

Short	Long	Example
`\,`	`\thinspace`	$\Rightarrow\!\Leftarrow$
`\:`	`\medspace`	$\Rightarrow\!\Leftarrow$
`\;`	`\thickspace`	$\Rightarrow\!\Leftarrow$
	`\quad`	$\Rightarrow\quad\Leftarrow$
	`\qquad`	$\Rightarrow\qquad\Leftarrow$
`\!`	`\negthinspace`	$\Rightarrow\!\!\!\!\Leftarrow$
	`\negmedspace`	$\Rightarrow\!\!\!\!\Leftarrow$
`\`	`\negthickspace`	$\Rightarrow\!\!\!\!\Leftarrow$

- More general, use `\mathspace{len}` with math units `mu` (1/18em)

Dots

- Different versions of dots:

`\dotsc`	dots with commas	$a, b, \ldots, z$
`\dotsb`	dots with binary operators	$1 + 2 + \cdots + n$
`\dotsm`	multiplication dots	$abc \cdots z$
`\dotsi`	dots with integrals	$\int_A \int_B \cdots$
`\dotso`	other dots	

  - Allows adaption of document to different conventions on the fly
- Dots in matrices over multiple rows with

```
\hdotsfor[spacing]{rows}
```

$$\begin{pmatrix} a & b & \ldots (\texttt{\textbackslash dots}) & c & d \\ \ldots & \ldots & \ldots & \ldots & (\texttt{\textbackslash hdotsfor[2]\{4\}}) \end{pmatrix}$$

Operator Names

- Add new math operator names like $\sin$:

```
\DeclareMathOperator{\xxx}{xxx}
```

  - Defines a new math operator commands `\xxx`
  - Proper spacing, textstyle, etc. is done automatically: you get $A \operatorname{xxx} B$ instead of $A\mathrm{xxx}B$ or even $Axxx B$
  - Only allowed in preamble
- Declare a math operator with subscripts and superscripts in *limit* positions:

```
\DeclareMathOperator*{\Lim}{Lim}
```

- For single use of operator name in formula:

```
\operatorname{abc}
```

Math Fonts

- Basic math font commands:
  `\mathbf{...}` `\mathrm{...}`
  `\mathcal{...}` `\mathsf{...}`
  `\mathtt{...}` `\mathit{...}`
- Additional fonts provided by `amsfonts`:
  Fraktur script: `\mathfrak{...}` $\mathfrak{Fraktur}$
  Symbols: `\mathbb{...}` $\mathbb{S}\curvearrowright\gg\bowtie\ll\sim:\mathbb{NZRC}$
- Boldface symbols:
  - \mathbf only does only work on latin letters
  - Get boldface symbols with `\boldsymbol{...}`

More Maths?

- A lot more math features are available
- Read the AMSLATEX documentation...