

The HEP-MATH package*

Extended math macros

Jan Hajer[†]

2026/08/01

Abstract

The HEP-MATH package extends MATHTOOLS and AMSMATH with HEP-oriented math notation. It provides math-only command definitions, improved overlines and arrow accents, operator names, fractions and unit helpers, differentials and derivative builders, paired delimiters, probabilities, brackets, quantum-mechanical bra-ket notation, matrix elements, and expectation values.

To use the package place `\usepackage{hep-math}` in the preamble.

The MATHTOOLS [1] package is loaded, which in turn loads the $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ AMSMATH [2] package. Horizontal spacing in inline equations and page breaks in block equations are marginally adjusted.

`\left` Spacing around `\left` and `\right` is fixed with the MLEFTRIGHT package [3].

`\right`

1 Macros

`\mathdef` The `\mathdef{<name>}{<arguments>}{<code>}` macro (re-)defines macros only within math mode without changing the text mode definition.

`\i` The imaginary unit `\i` and the differential `\d` are defined using this functionality.

`\d` The `\overline` macro is adjusted to work also outside of math mode using the SOUL [4] package.

`\overline` A better looking over left right arrow is defined i.e. $\overrightarrow{\partial}$ using a new `\oset{<over>}{<math>}` functionality.

`\oset`

`\overleftarrow` Diagonal matrix `\diag`, signum `\sgn`, trace `\tr`, `\Tr`, and `\rank` operators are defined.

`\overrightarrow` The real and imaginary projectors are redefined to look like ordinary operators.

`\overleftrighth` `\cos` and `\tan` are adjusted to have the same height as `\sin`.

`\diag` `\arccsc` and other inverse trigonometric functions are defined.

`\sgn`

`\Re` 1.1 Fractions and units

`\Im` The correct spacing for units is provided by the macro `\unit{<value>}{<unit>}` from the UNITS package [5] which can also be used in text mode. The macro `\inv{<power>}{<text>}` allows to avoid math mode also for inverse units such as 5fb^{-1} typeset via `\unit[5]{\inv{fb}}`.

`\sin`

`\cos` The `\frac{<number>}{<number>}` macro is accompanied by `\nicefrac{<number>}{<number>}`,

`\tan`

*This document corresponds to HEP-MATH v1.5.

[†]jan.hajer@tecnico.ulisboa.pt

`\accsc`

`\unit`

`\inv`

`\nicefrac`

`\flatfrac`

`\textfrac`

`\textfrac{⟨number⟩}{⟨number⟩}`, and `\flatfrac{⟨number⟩}{⟨number⟩}` leading to $\frac{1}{2}$, $\frac{1}{2}$, $\frac{1}{2}$, and $\frac{1}{2}$. The `\textfrac` macro is mostly intended if a font with oldstyle numerals is used.

Some macros of the PHYSICS package [6] are reimplemented with a more conventional typesetting in mind. Finer details about mathematical typesetting can be found in [7].

1.2 Differentials and derivatives

`\differential` The three macros `\differential{⟨symbol⟩}`, `\newderivative{⟨name⟩}{⟨symbol⟩}`, and `\newpartialderivative{⟨name⟩}{⟨symbol⟩}` allow to define a differential with correct spacing, a derivative using this differential, and if necessary a partial derivative that can handle three dimensional derivatives.

`\newpartialderivative` These macros are used for the usual differential and derivative, producing dx via `\d x` and

<code>\d</code>	<code>\dv[f]x</code>	<code>\dv*[f]x^n</code>	<code>\dv[f]x*x^n</code>	<code>\dv*[f]x*x^n</code>
<code>\dv</code>	$\frac{df}{dx}$	$d^n f / dx^n$	$\frac{d^n f}{dx^n}$	$d^n f / dx^n$
	<code>\dv x f</code>	<code>\dv*x f</code>	<code>\dv x*f</code>	<code>\dv*x*f</code>
	$\frac{d}{dx} f$	$d/dx f$	$\frac{d}{dx} f$	$d/dx f$

via `\dv*[⟨f⟩]{⟨x⟩}*^{⟨n⟩}`. Upright differential can be produced via `\renewcommand{\diffsymbol}{\mathrm d}`. The differential takes care of the correct spacing as long as it is placed at the end of the integral $\int f(x) dx$. In order to archive correct spacing when it is placed at the beginning of the integral it is advisable to place the whole expression in a `\mathop{\int\limits\mathrm d x} f(x)` such that $\int dx f(x)$.

`\pd` Similarly a partial differential and derivative are defined that can be used according to `\pdv*[⟨f⟩]{⟨x⟩}*^{⟨a⟩}[⟨y⟩]^{⟨b⟩}[⟨z⟩]^{⟨c⟩}`.

<code>\pdv[f]x</code>	<code>\pdv[f]x[y]</code>	<code>\pdv[f]x^3</code>	<code>\pdv[f]x^2[y]</code>
$\frac{\partial f}{\partial x}$	$\frac{\partial^2 f}{\partial x \partial y}$	$\frac{\partial^3 f}{\partial x^3}$	$\frac{\partial^3 f}{\partial x^2 \partial y}$
<code>\pdv[f]x^2[y]^3</code>	<code>\pdv[f]x[y]^3</code>	<code>\pdv x[y]f</code>	
$\frac{\partial^5 f}{\partial x^2 \partial y^3}$	$\frac{\partial^4 f}{\partial x \partial y^3}$	$\frac{\partial^2}{\partial x \partial y} f$	

`\var` Similarly a functional variation and functional derivative are defined.

`\fdv` The `\cancel{⟨characters⟩}` macro from the CANCEL package [8] and the `\slashed{⟨character⟩}` macro from the SLASHED package [9] allow to ~~cancel~~ math and use the Dirac slash notation i.e. $\cancel{\emptyset}$, respectively.

1.3 Paired delimiters

`\abs`

`\norm` `\abs x` `\norm x` `\norm[2]x` `\norm*[2]x`
 $|x|$ $\|x\|$ $\|x\|_2$ $\|x\|_2$

`\eval`

`\order` `\order x` `\eval x_0^{\infty}` `\eval* x_0^{\infty}`
 $\mathcal{O}(x)$ $x|_0^\infty$ $x|_0^\infty$

`\newpair` The `\newpair{⟨name⟩}{⟨left delim⟩}{⟨right delim⟩}_{⟨subscript⟩}^{⟨superscript⟩}` macro is defined and used for the definition of (anti-)commutators and Poisson brackets.

`\comm`

`\acomm`

$$\begin{array}{ccc} \backslash pb\ xy & \backslash comm\ xy & \backslash acomm\ xy \\ \{x,y\} & [x,y] & \{x,y\} \end{array}$$

They can easily be redefined using e.g. `\newpair\comm\lbrack\rbrack_-`.

<code>\bra</code>	Macros for the bra-ket notation are introduced.			
<code>\ket</code>	<code>\bra x</code>	<code>\ket x</code>	<code>\braket xy</code>	<code>\ketbra xy</code>
	$\langle x $	$ x \rangle$	$\langle x y \rangle$	$ x \rangle \langle y $
<code>\braket</code>	<code>\mel xyz</code>	<code>\ev x</code>	<code>\ev[\Omega] x</code>	<code>\vev x</code>
	$\langle x y z \rangle$	$\langle x \rangle$	$\langle \Omega x \Omega \rangle$	$\langle 0 x 0 \rangle$
<code>\ketbra</code>				
<code>\mel</code>	Macros for row and column vectors are introduced together with a symbol for transpose vectors.			
<code>\ev</code>	<code>\column{x,y,z}</code>	<code>\row{x,y,z}^{\sim trans}</code>		
<code>\vev</code>	$\begin{pmatrix} x \\ y \\ z \end{pmatrix}$	$(x, y, z)^T$		
<code>\column</code>				
<code>\row</code>				

2 Environments

`eqnarray` The `eqnarray` environment is depreciated, the `split`, `multline`, `align`, `multlined`, `aligned`, `alignedat`, and `cases` environments of the AMSMATH and MATHTOOLS packages should be used instead.

`equation` Use the `equation` environment for short equations.
`\begin{equation}`
`left = right \ .` $\boxed{\text{left}} = \boxed{\text{right}} . \quad (2.1)$
`\end{equation}`

`multline` Use the `multline` environment for longer equations.
`\begin{multline}`
`left = right 1 \\
+ right 2 \ .` $\boxed{\text{left}} = \boxed{\text{right 1}} + \boxed{\text{right 2}} . \quad (2.2)$
`\end{multline}`

`split` Use the `split` sub environment for equations in which multiple equal signs should be aligned.
`\begin{equation} \backslash begin{split}`
`left \&= right 1 \\
\&= right 2 \ .` $\boxed{\text{left}} = \boxed{\text{right 1}} = \boxed{\text{right 2}} . \quad (2.3)$
`\end{split} \end{equation}`

`align` Use the `align` environment for the vertical alignment and horizontal distribution of multiple equations.
`\begin{subequations} \backslash begin{align}`
`left \&= right \ , \&
left \&= right \ , \\
left \&= right \ , \&
left \&= right \ .` $\boxed{\text{left}} = \boxed{\text{right}} , \quad \boxed{\text{left}} = \boxed{\text{right}} , \quad (2.4a)$
 $\boxed{\text{left}} = \boxed{\text{right}} , \quad \boxed{\text{left}} = \boxed{\text{right}} . \quad (2.4b)$
`\end{align} \end{subequations}`

`aligned` Use the `aligned` environment within a `equation` environment if the aligned equations should be labeled with a single equation number.

`multlined` Use the `multlined` environment if either `split` or `align` contain very long lines.

```

\begin{equation} \begin{split}
left \&= right 1 \&=
\begin{multlined}[t]
right 2 \&+ right 3 \ .
\end{multlined}
\end{split} \end{equation}

```

$$\begin{aligned}
\boxed{\text{left}} &= \boxed{\text{right 1}} \\
&= \boxed{\text{right 2}} \\
&\quad \boxed{+ \text{right 3}} .
\end{aligned} \tag{2.5}$$

alignat Use the `alignat` environment together with the `\mathllap` macro for the alignment of multiple equations with vastly different lengths.

```

\begin{subequations}
\begin{alignat}{2}
left \&= long right \&\ , \&
le. 2 \&= ri. 2 \ , \&
\mathllap{le. 3 = ri. 3} \&\ .
\end{alignat}
\end{subequations}

```

$$\boxed{\text{left}} = \boxed{\text{long right}} , \tag{2.6a}$$

$$\boxed{\text{le. 2}} = \boxed{\text{ri. 2}} , \quad \boxed{\text{le. 3}} = \boxed{\text{ri. 3}} . \tag{2.6b}$$

As a rule of thumb if you have to use `\notag`, `\nonumber`, or perform manual spacing via `\quad` you are probably using the wrong environment.

A Implementation

<*package>

Load the `MATHTOOLS` package [1] which loads the `AMSMATH` package [2]. Allow page breaks within equations if necessary. Adjust the thick and med mu skips slightly.

```

1 \RequirePackage{mathtools}
2 \mathtoolsset{centercolon}
3 \numberwithin{equation}{section}
4 \allowdisplaybreaks[1]
5 \thickmuskip=5mu plus 3mu minus 1mu
6 \medmuskip=4mu plus 2mu minus 3mu

```

\mathdef Define the `\mathdef{<name>}[<arguments>]{<macro>}` macro which (re-)defines macros in math mode only. This macro is implemented using the `XPARSE` package [10].

```

7 \RequirePackage{xparse}
8 \DeclareDocumentCommand{\mathdef}{mO{0}om}{%
9   \expandafter\let\csname hep@text\string#1\endcsname=#1
10  \expandafter\newcommand\expandafter*\csname hep@math\string#1%
11  \IfNoValueTF{#3}{\endcsname[#2]}{\endcsname[#2] [#3]}{#4}
12  \DeclareRobustCommand#1{%
13    \ifmmode
14      \expandafter\let\expandafter\next\csname%
15      hep@math\string#1\endcsname%
16    \else
17      \expandafter\let\expandafter\next\csname%
18      hep@text\string#1\endcsname%
19    \fi
20    \next
21  }%

```

22 }

`\i` Provide an upright imaginary unit in math mode.

```
23 \newcommand*\imaginaryunit{i}
24 \AtBeginDocument{\mathdef{i}{\imaginaryunit}}
```

`\overline` Redefine `\overline` to be a text macro using the SOUL package [4]. Extend it as a math macro with the original definition from the AMSMATH package [2].

```
25 \RequirePackage{soul}
26 % \def\overline#1{{\renewcommand*\ULdepth{-1.9ex}}{\uline{#1}}}
27 \newcommand*{\textoverline}[1]{{\setul{-1.9ex}}{\ul{#1}}}
28 \let\overline\textoverline
29 \DeclareRobustCommand{\over@line}[1]{\@@overline{#1}}
30 \mathdef{\overline}{\over@line}
31 \newcommand*\hep@widebar[1]{%
32   \mkern2.5mu\overline{\mkern-2.5mu#1\mkern-.5mu}\mkern.5mu%
33 }
34 \newcommand*\widebar[1]{%
35   \settowidth{\dimen0}{\ensuremath{#1}}%
36   \ifdim\dimen0>.475em\hep@widebar{#1}\else\bar{#1}\fi%
37 }
```

`\oset` Define a new overset macro `\oset[⟨offset⟩]{⟨over⟩}{⟨base⟩}`

```
38 \newcommand*\oset{[3] [-1pt]}%
39 \text{\raisebox{.2ex}{\mathop{#3}\limits^{%
40   \vbox to#1{\kern-2\ex@\hbox{\scriptscriptstyle#2$}\vss}%
41   }$}}}%
42 }
```

`\overleftarrow` Define a over left right arrow `\overleftarrow{⟨base⟩}`.

```
43 \newcommand*\overleft{[1]{\oset{\leftarrow}{#1}}}
44 \newcommand*\overrightarrow{[1]{\oset{\rightarrow}{#1}}}
45 \newcommand*\overleftright{[1]{\oset{\leftrightarrow}{#1}}}
```

`eqnarray` Undefine the `eqnarray` environment if not prevented by package option.

```
46 % \newif\ifhep@eqnarray\hep@eqnarraytrue
47 % \ifhep@eqnarray\else
48 %   \let\eqnarray\@undefined
49 %   \let\endeqnarray\@undefined
50 % \fi
```

A.1 Operators

`\tr` Provide the `\diag`, `\sgn`, and some other operators.

```
\Tr
\rank 51 \DeclareMathOperator{\tr}{tr}
\erf   52 \DeclareMathOperator{\Tr}{Tr}
\Res
\sgn
\sgn
\diag
```

```

53 \DeclareMathOperator{\rank}{rank}
54 \DeclareMathOperator{\erf}{erf}
55 \DeclareMathOperator{\Res}{Res}
56 \DeclareMathOperator{\sgn}{sgn}
57 \DeclareMathOperator{\diag}{diag}
58 \let\det\relax\DeclareMathOperator{\det}{det}

```

`\Re` Redefine the real and imaginary projectors.

```

\Im
59 \let\Re\relax\DeclareMathOperator{\Re}{Re}
60 \let\Im\relax\DeclareMathOperator{\Im}{Im}

```

`\transpose` Define a transpose symbol.

```

\trans
61 \RequirePackage{amssymb}
62 \newcommand*{\hep@transpose}[2]{\raisebox{\depth}{$\math@th#1\intercal$}}
63 \newcommand*{\transpose}{\mathpalette\hep@transpose{}}
64 \let\trans\transpose

```

A.1.1 Trigonometric functions

`\cos` Adjust the height of `\cos` and `\tan` to be equal to `\sin`.

```

\atan
65 \let\cos\undefined\DeclareMathOperator{\cos}{\cos\vphantom{i}}
66 \let\tan\undefined\DeclareMathOperator{\tan}{\tan\vphantom{i}}

```

`\arccsc` Define arc operators.

```

\arcsec
\arccot
67 \DeclareMathOperator{\arccsc}{arccsc}
68 \DeclareMathOperator{\arcsec}{arcsec}
69 \DeclareMathOperator{\arccot}{arccot}

```

`\asin` Define shorthand for arc operators.

```

\acos
\atan
70 \DeclareMathOperator{\asin}{asin}
\acsc
71 \DeclareMathOperator{\acos}{acos}
\asec
72 \DeclareMathOperator{\atan}{atan}
\acot
73 \DeclareMathOperator{\acsc}{acsc}
74 \DeclareMathOperator{\asec}{asec}
75 \DeclareMathOperator{\acot}{acot}

```

`\csch` Define `\csch` and `\sech` operators.

```

\sech
76 \DeclareMathOperator{\csch}{csch}
77 \DeclareMathOperator{\sech}{sech}

```

A.2 Units and fractions

`\nicefrac` Define the `\nicefrac` macro using the `xfrac` package [11].

```

78 \RequirePackage{xfrac}
79 \let\nicefrac\sfrac

```

Load the SIUNITX package [12].

```

80 \RequirePackage{siunitx}
81 \providecommand*\hep@lining@style{\ifdefined\lstyle\lstyle\fi}
82 \sisetup{
83   reset-math-version=false,
84   reset-text-family=false,
85   reset-text-series=false,
86   uncertainty-mode=separate,
87   exponent-product=\cdot,
88   unit-mode=text,
89   text-font-command=\hep@lining@style,
90 }

```

Provide a compatibility macro for older installations.

```

91 \ExplSyntaxOn
92 \cs_if_exist:NF \siunitx_declare_prefix:Nne
93 {
94   \cs_generate_variant:Nn
95   \siunitx_declare_prefix:Nnn
96   { Nne }
97 }
98 \ExplSyntaxOff

```

`\unit` Ensure that its macros are compatible with the definitions used in the depreciated `UNIT` package.
`\unitfrac`

```

99 \NewCommandCopy{\hep@unit}{\unit}
100 \ExplSyntaxOn
101 \siunitx_declare_prefix:Nne\mu{-6}{\_siunitx_unit_non_latin:n{"03BC}}
102 \prg_new_conditional:Npnn\hep_if_keyval:n#1{TF}{
103   \tl_if_in:nnTF{#1}{=}{\prg_return_true:}{\prg_return_false:}
104 }
105 \cs_new_protected:Npn\hep_unit:nn#1#2{
106   \hep_if_keyval:nTF{#1}{\hep@unit{#1}{#2}}{\qty[parse-numbers=false]{#1}{#2}}
107 }
108 \RenewDocumentCommand{\unit}{om}{
109   \IfNoValueTF{#1}{\hep@unit{#2}}{\hep_unit:nn{#1}{#2}}
110 }
111 \ExplSyntaxOff
112 \NewDocumentCommand{\unitfrac}{omm}{
113   \IfNoValueTF{#1}{\hep@unit{#2\per#3}}{\qty{#1}{#2\per#3}}
114 }

```

`\inv` Provide a macro for the inverse, useful in combination with the unit macro in text mode.

```

115 \newcommand*\inv[2][1]{#2\ensuremath{\sim^{-#1}}}

```

`\textfrac` Provide the `\textfrac` macro useful in combination with a font using lining numerals.

```

116 \newcommand*\textfrac[2]{\ensuremath{\nicefrac{\text{#1}}{\text{#2}}}}

```

`\flatfrac` Provide a flat fraction.

```

117 \DeclarePairedDelimiterX{\hep@flatfrac}[2]{.}{.}{.}{.}{%
118   \kern-\nulldelimiterspace#1\delimspace/%
119   \hep@left@delim#2\kern-\nulldelimiterspace%
120 }
121 \NewDocumentCommand{\flatfrac}{somm}{%
122   \mathinner{%
123     \IfBooleanTF{#1}{%
124       \hep@flatfrac*{#3}{#4}%
125     }{%
126       \IfNoValueTF{#2}{\hep@left@delim#3/\hep@left@delim#4%
127       }{%
128         \hep@flatfrac[#2]{#3}{#4}%
129       }%
130     }%
131   }%
132 }

```

A.2.1 Differentials and derivatives

`\int` Redefine `\int`.

```

133 % \let\hep@int\int
134 % \RenewDocumentCommand{\int}{oe{~}}{%
135 %   \def\temp{\hep@int_{\IfValueT{#2}{#2}}^{\IfValueT{#3}{#3}}}%
136 %   \IfValueTF{#1}{\mathop{\temp#1}}{\temp}%
137 % }

```

`\differential` Define a generic differential `\differential`.

```

138 \newcommand*{\differential}[1]{\mathop{}}\!#1}

```

`\newderivative` Define a generic derivative.

```

139 \newcommand*\newderivative[2]{
140   \NewDocumentCommand{#1}{somse{~}}{%
141     \IfBooleanTF{##4}{%
142       \IfBooleanTF{##1}{\nicefrac}{\frac}%
143     }{%
144       \IfBooleanTF{##1}{\flatfrac}{\dfrac}%
145     }{%
146       \differential#2\IfValueT{##5}{~{##5\!}}\IfValueT{##2}{##2}%
147     }{%
148       \differential#2{~}{##3}\IfValueT{##5}{~{##5}}%
149     }%
150   }
151 }

```

`\newpartialderivative` Define a generic partial derivative

```

152 \newcommand*\newpartialderivative[2]{

```

```

153 \NewDocumentCommand{#1}{somsE{~}{1}oE{~}{1}oE{~}{1}}{%
154 \def\hep@one{\IfValueTF{##6}{##7}{0}}
155 \def\hep@two{\IfValueTF{##8}{##9}{0}}
156 \def\hep@sum{\the\numexpr##5+\hep@one+\hep@two\relax}
157 \IfBooleanTF{##4}{%
158 \IfBooleanTF{##1}{\nicefrac}{\frac}%
159 }{%
160 \IfBooleanTF{##1}{\flatfrac}{\dfrac}%
161 }{%
162 \differential#2\ifnum\hep@sum=1\relax\else{~\hep@sum!}\fi
163 \IfValueT{##2}{##2}%
164 }{%
165 \differential#2{##3}\ifnum##5=1\relax\else{~{##5}}\fi%
166 \IfValueT{##6}{#2##6\ifnum##7=1\relax\else{~{##7}}\fi}%
167 \IfValueT{##8}{#2##8\ifnum##9=1\relax\else{~{##9}}\fi}%
168 }%
169 }
170 }

```

`\diffsymbol` Define the differential `\d` and the usual derivative.

```

\diff
\d 171 \providecommand*{\diffsymbol}{d}
\derivative 172 \newcommand*{\diff}{\differential\diffsymbol}
\dv 173 \AtBeginDocument{\mathdef{\d}{\diff}}
174 \newderivative{\derivative}{\diffsymbol}
175 \newcommand*\dv{\derivative}

```

`\partialdifferential` Define the partial differential and derivative.

```

\pd
\partialderivative 176 \newcommand*\partialdifferential{\differential\partial}
\pdv 177 \newcommand*\slashedpartialdifferential{\differential{\slashed\partial}}
178 \newcommand*\spd{\slashedpartialdifferential}
179 \newcommand*\pd{\partialdifferential}
180 \newpartialderivative{\partialderivative}{\partial}
181 \newcommand*\pdv{\partialderivative}

```

`\gaugediffsymbol` Define the gauge covariant differential `\D`.

```

\gaugediff
\D 182 \providecommand*{\gaugediffsymbol}{D}
183 \newcommand*{\gaugediff}{\differential\gaugediffsymbol}
184 \newcommand*{\D}{\gaugediff}

```

`\covariantdiff` Define the covariant differential `\cd`.

```

\cd
185 \newcommand*{\covariantdiff}{\differential\nabla}
186 \newcommand*{\cd}{\covariantdiff}

```

`\variation` Define the functional variation and derivative.

```

\var
functionalderivative 187 \newcommand*\variation{\differential\delta}
\dvdv 188 \newcommand*\var{\variation}
189 \newpartialderivative{\functionalderivative}{\delta}

```

190 `\newcommand*\fdv{\functionalderivative}`

`\cancel` Load the CANCEL [8] and SLASHED [9] packages which provide the `\cancel` and `\slashed` macros.
`\slashed`

```
191 \RequirePackage{cancel}
192 \RequirePackage{slashed}
193 \declareslashed{}{/}{.14}{0}{L}
194 \declareslashed{}{/}{.055}{0}{\partial}
195 \declareslashed{}{/}{.06}{0}{\gaugediff}
196 \declareslashed{}{/}{.055}{0}{\partialdifferential}
```

A.3 Paired delimiters

`\left` Load the MLEFTRIGHT package [3] and adjust the spacing around `\left` and `\right`.
`\right`

```
197 \RequirePackage{mleftright}
198 \mleftright
```

`\noargumentsymbol` Allow for macros to have an empty argument using the ETOOLBOX package [13].
`\optionalargument`

```
199 \RequirePackage{etoolbox}
200 \newcommand*\noargumentsymbol{\:\cdot\:}
201 \newcommand*\optionalargument[1]{\ifblank{#1}{\noargumentsymbol}{#1}}
```

`\abs` Absolute value and norm.

```
\norm
202 \DeclarePairedDelimiterX\abs[1]\lvert\rvert{\optionalargument{#1}}
203 \DeclarePairedDelimiterX\hep@norm[1]\lVert\rVert{\optionalargument{#1}}
204 \DeclarePairedDelimiterXPP\hep@pnorm[2]{}\lVert\rVert{_{#1}}{#2}
205 \NewDocumentCommand{\norm}{som}{%
206   \IfValueTF{#2}{%
207     \IfBooleanTF{#1}{\hep@pnorm*}{\hep@pnorm}{#2}%
208   }{%
209     \IfBooleanTF{#1}{\hep@norm*}{\hep@norm}%
210   }\optionalargument{#3}%
211 }
```

`\floor` Floor and ceiling paired delimiters.

```
\ceil
212 \DeclarePairedDelimiter\ceil{\lceil}{\rceil}
213 \DeclarePairedDelimiter\floor{\lfloor}{\rfloor}
```

`\ordersymbol` Order symbol and macro.

```
\order
214 \providecommand*\ordersymbol{\mathcal{O}}
215 \DeclarePairedDelimiterXPP\order[1]{\ordersymbol}{}{#1}
```

`\evaluated` Vertical evaluation bar

```
\eval
216 \DeclarePairedDelimiter{\hep@evaluated}{.}{\rvert}
217 \NewDocumentCommand{\evaluated}{som}{%
218   \IfBooleanTF{#1}{%
```

```

219     \hep@evaluated*{#3}%
220   }{%
221     \IfNoValueTF{#2}{#3\rvrt}{\hep@evaluated[#2]{#3}}%
222   }%
223 }
224 \newcommand*\eval{\evaluated}

\row Shortcuts for rows and columns
\column
225 \newcommand*\rowseparator{,\,}
226 \ExplSyntaxOn
227 \newcommand*\hep@row[1]{
228   \seq_set_split:Nnn\hep@seq{,}{#1}
229   \begin{matrix}\seq_use:Nn\hep@seq{\rowseparator}\end{matrix}
230 }
231 \newcommand*\hep@column[1]{%
232   \seq_set_split:Nnn\hep@seq{,}{#1}%
233   \begin{matrix}\seq_use:Nn\hep@seq{\,}\end{matrix}%
234 }
235 \ExplSyntaxOff
236 \DeclarePairedDelimiterX{\hep@row@}[1]{(}{)}{\hep@row{#1}}
237 \NewDocumentCommand{\hep@column@}{me{~}e_{}}{%
238   \left(\hep@column{#1}\right)%
239   \IfValueT{#2}{~\!\!\!#2}\IfValueT{#3}{_{\!\!\!#3}}%
240 }
241 \providecommand*\column{}
242 \providecommand*\row{}
243 \mathdef{\column}{\hep@column@}
244 \providecommand*\row{}
245 \mathdef{\row}{\hep@row@}

```

A.3.1 Set and Probability

`\midbar` Define a generic midbar.

```

246 \newcommand*\hep@left@delim{\mathopen{}}
247 \providecommand*\midbar[1][]{%
248   \nonscript\:#1\vert\allowbreak\nonscript\:\hep@left@delim%
249 }

```

Check if `nfssect-cfr` is loaded and patch the global `\set` macro into the `cfr` namespace

```

250 \RequirePackage{xpatch}
251 \@ifundefined{exfs@merge@families}{}{%
252   \xpatchcmd{\exfs@merge@families}{\set}{\cfr@set}{}{}%
253   \xpatchcmd{\exfs@merge@families}{\set}{\cfr@set}{}{}%
254   \xpatchcmd{\exfs@merge@families}{\set}{\cfr@set}{}{}%
255 }%

```

`\suchthat` Define a `\set` macro that allows a midbar via `\suchthat`.

```

\set
256 \providecommand*\suchthat{\midbar}

```

```

257 \DeclarePairedDelimiterX\set[1]{\}\}{\}%
258 \renewcommand*\suchthat{\midbar[\delimsize]}#1%
259 }

```

`\probabilitysymbol` Redefine the `\Pr` macro to a macro that takes a `\given` macro and generates a midbar.

```

\given
\Pr 260 \providecommand*\probabilitysymbol{\operatorname{Pr}}
261 \providecommand*\given{\midbar}
262 \DeclarePairedDelimiterXPP\hep@Pr[1]{\%
263 \probabilitysymbol}{\}\}{\}%
264 \renewcommand*\given{\midbar[\delimsize]}#1%
265 }
266 \let\Pr\relax
267 \NewDocumentCommand{\Pr}{so}{\%
268 \IfValueTF{#2}{\%
269 \IfBooleanTF{#1}{\hep@Pr*}{\hep@Pr}{#2}%
270 }{\%
271 \probabilitysymbol%
272 }%
273 }

```

A.3.2 Commutators

`\newpair` Define the `\newpair` macro that generates pairs surrounded by brackets.

```

274 \NewDocumentCommand{\newpair}{mmme{_{}}e{^}}{\%
275 \IfNoValueTF{#4}{\%
276 \IfNoValueTF{#5}{\%
277 \DeclarePairedDelimiterX{#1}[2]{#2}{#3}%
278 }{\%
279 \DeclarePairedDelimiterXPP{#1}[2]{\}{#2}{#3}{^#5}}%
280 }%
281 }{\%
282 \DeclarePairedDelimiterXPP{#1}[2]{\}{#2}{#3}{_{#4}}}%
283 }{\%
284 \optionalargument{##1},\optionalargument{##2}%
285 }%
286 }

```

`\innerproduct` Poissonbracket, commutator and anti-commutator.

```

\poissonbracket
\pb 287 \newpair\innerproduct\langle\rangle
\commutator 288 \newpair\poissonbracket\lbrace\rbrace
\comm 289 \newpair\commutator\lbrack\rbrack
\anticommutator 290 \newcommand*\pb{\poissonbracket}
291 \newcommand*\comm{\commutator}
292 \newcommand*\acomm{\poissonbracket}

```

A.3.3 Bra-ket notation

`\bracketspace` Define the space within bracket notation.

```

293 % \providecommand*\braketspace{\mskip1mu}
294 \providecommand*\braketouterspace{\mskip1mu}
295 \providecommand*\braketinnerspace{\mskip3mu}
296 \newcommand*\hep@midvert{%
297   \braketinnerspace\delimsizes\vert\braketinnerspace\hep@left@delim%
298 }

```

`\braket` Define the `braket` macro.

```

299 \DeclarePairedDelimiterX\braket[2]{\langle}{\rangle}{%
300   \braketouterspace#1\hep@midvert#2\braketouterspace%
301 }

```

`\bra` Define the `bra` macro.

```

302 \DeclarePairedDelimiterXPP\hep@bra[1]{%
303   }{\rangle}{\rvert}{\braketinnerspace%
304   }{\braketouterspace#1\braketinnerspace%
305 }
306 \NewDocumentCommand{\bra}{smt\ket sgt\ketbra sgg}{%
307   \IfBooleanTF{#6}{%
308     \IfBooleanTF{#1}{\braket*{#2}{#8}}{\braket{#2}{#8}}%
309     \IfBooleanTF{#7}{\bra*{#9}}{\bra{#9}}%
310   }{
311     \IfBooleanTF{#3}{%
312       \IfBooleanTF{#1}{\braket*}{%
313         \IfBooleanTF{#4}{\braket*}{\braket}}{#2}{#5%
314       }%
315     }{
316       \IfBooleanTF{#1}{\hep@bra*}{\hep@bra}{#2}%
317     }%
318   }%
319 }

```

`\ket` Define the `ket` macro.

```

320 \DeclarePairedDelimiterXPP\ket[1]{%
321   \braketinnerspace}{\lvert}{\rangle}{%
322 }{%
323   \braketinnerspace\hep@left@delim#1\braketouterspace%
324 }

```

`\ketbra` Define the `ketbra` macro.

```

325 \NewDocumentCommand{\ketbra}{smm}{%
326   \IfBooleanTF{#1}{%
327     \ket*{#2}\bra*{#3}%
328   }{
329     \ket{#2}\bra{#3}%
330   }%
331 }

```

```

\matricelement Define the matricelement macro.
\mel
332 \DeclarePairedDelimiterX\matricelement[3]{%
333   \langle\rangle
334 }{%
335   \braketouterspace#1\hep@midvert#2\hep@midvert#3\braketouterspace%
336 }
337 \newcommand*\matrixel{\matricelement}
338 \newcommand*\mel{\matricelement}

\expectationvalue Define the expectationvalue and vev macros.
\ev
\vev 339 \DeclarePairedDelimiterX\hep@expvalue[1]{\langle\rangle}%
340   \braketouterspace#1\braketouterspace%
341 }
342 \NewDocumentCommand{\expectationvalue}{som}{%
343   \IfNoValueTF{#2}{%
344     \IfBooleanTF{#1}{\hep@expvalue*}{\hep@expvalue}{#3}%
345   }{%
346     \IfBooleanTF{#1}{\matricelement*}{\matricelement}{#2}{#3}{#2}%
347   }%
348 }
349 \newcommand*\ev{\expectationvalue}
350 \newcommand*\vev[1]{\expectationvalue[0]{#1}}

</package>

```

B Test

```

<*test>

351 \documentclass{article}
352
353 \usepackage{hep-math}
354
355 \begin{document}
356
357 \begin{gather}
358   \bra{x}\ket{y}
359   \braket*{x}{y}\backslash
360   \dv[f]{x}^3
361   \pdv[f]{x}[y]^2[z]^3
362   \fdv[f]{x}^3[y]\backslash
363   \set{x \suchthat x \in X}
364 \end{gather}
365
366 \end{document}
367

</test>

```

C Readme

<*readme>

```
368 # The 'hep-math' package
369
370 Extended math macros
371
372 ## Introduction
373
374 The 'hep-math' package extends 'mathtools' and 'amsmath' with
375 HEP-oriented math notation. It provides math-only command definitions,
376 improved overlines and arrow accents, operator names, fractions and unit
377 helpers, differentials and derivative builders, paired delimiters,
378 probabilities, brackets, quantum-mechanical bra-ket notation, matrix
379 elements, and expectation values.
380
381 To use the package place '\usepackage{hep-math}' in the preamble.
382
383 ## Author
384
385 Jan Hajer
386
387 ## License
388
389 This file may be distributed and/or modified under the conditions of the
390 'LaTeX' Project Public License, either version 1.3c of this license or
391 (at your option) any later version. The latest version of this license is
392 in 'http://www.latex-project.org/lppl.txt' and version 1.3c or later is
393 part of all distributions of LaTeX version 2005/12/01 or later.
```

</readme>

References

- [1] L. Madsen, M. Høgholm, W. Robertson, and J. Wright. 'The `mathtools` package: Mathematical tools to use with `amsmath`' (2004). CTAN: `mathtools`.
- [2] *LaTeX Team*. 'The `amsmath` package: AMS mathematical facilities for $\LaTeX$ ' (1994). CTAN: `amsmath`. URL: ams.org/tex/amslatex.
- [3] H. Oberdiek. 'The `mleftright` package: Variants of delimiters that act as maths open/close' (2010). CTAN: `mleftright`.
- [4] M. Franz. 'The `soul` package: Hyphenation for letterspacing, underlining, and more' (1998). CTAN: `soul`.
- [5] A. Reichert. 'The `units` and `nicefrac` packages: Typeset units' (1998). CTAN: `units`.
- [6] S. C. de la Barrera. 'The `physics` package: Macros supporting the Mathematics of Physics' (2012). CTAN: `physics`.
- [7] E. Gregorio. 'TeX, LaTeX and math' (2020). URL: latex-project.org/publications/2020-egreg-TUB-tbl27gregorio-math.pdf.
- [8] D. Arseneau. 'The `cancel` package: Place lines through maths formulae' (2013). CTAN: `cancel`.

- [9] D. Carlisle. ‘The `slashed` package: Put a slash through characters’ (1987). CTAN: `slashed`.
- [10] *L^AT_EX₃ Project*. ‘The `xparse` package: A generic document command parser’ (1999). CTAN: `xparse`.
- [11] *L^AT_EX₃ Project*. ‘The `xfrac` package: Split-level fractions’ (2004). CTAN: `xfrac`.
- [12] J. Wright. ‘The `siunitx` package: A comprehensive (SI) units package’ (2008). CTAN: `siunitx`.
- [13] P. Lehman and J. Wright. ‘The `etoolbox` package: e- $\TeX$ tools for L^AT_EX’ (2007). CTAN: `etoolbox`.

Change History

v1.0	General: Created the dedicated math helper package with <code>\mathdef</code> , overlines, accents, operators, fractions, units, differentials, delimiters, probabilities, brackets, bra-ket notation, matrix elements, and vectors.	1
		<code>\floor</code> and <code>\ceil</code> , and differential and flat-fraction spacing refinements.
		v1.3
		General: Switched underline support from <code>soulutf8</code> to <code>soul</code> , disabled the active <code>\int</code> redefinition, added <code>\slashedpartialdifferential</code> and <code>\spd</code> , and exposed <code>\row</code> and <code>\column</code> through <code>\mathdef</code>
		1
v1.1	General: Added optional defaults to <code>\mathdef</code> , <code>\widebar</code> , row and column vector documentation, centred colons, <code>amssymb</code> loading, improved flat fractions and slashed symbols, <code>cfr-lm</code> and <code>nfssect-cfr</code> patching, and separate bra-ket spacing.	1
		v1.4
		General: Applied starred command definitions consistently and refreshed release metadata.
		1
		v1.5
		General: Raised the LaTeX format requirement, bumped release metadata, and expanded the documentation, README, and examples for the math interface.
		1
v1.2	General: Added <code>\imaginaryunit</code> , math-mode <code>\unit</code> and <code>\unitfrac</code> wrappers, optional integral placement,	