MathSemantics.sty – Semantic Math Commands

Ronny Bergmann
ronny.bergmann@ntnu.no
Department of Mathrmatical Sciences
NTNU, Trondheim, Norway

Roland Herzog
roland.herzog@iwr.uni-heidelberg.de
Interdisciplinary Center for Scientific Computing
Heidelberg University, Germany

2022-05-05

Contents

1 Introduction

This package aims to provide semantic commands for ease of use in mathematics to see better *what* you semantically mean which should be distinct/split from *how* it is realised in LaTeX.

The package is a spin-off and developed in the suite of packages from the former numapde-group in Chemnitz, see the original repository at `https://gitlab.hrz.tu-chemnitz.de/numapde-public/numapde-latex`.

Throughout this documentation most commands are directly illustrated by examples, which are both displayed as code (</> or </>$ for math examples) and its rendered result in LaTeX (👁). Two examples are
</>$ `\bbR` 👁 $\mathbb{R}$
and
</> `\eg` 👁 e. g.

The aim is to first ease the use of some often used letters and low-level formats like bold face letters </>$ `\bbR` 👁 $\mathbb{R}$, but also to provide high level commands that make typing mathematics easier, for example using </>$ `\abs{\frac{1}{2}}` 👁 $|\frac{1}{2}|$ and </>$ `\abs[Big]{\frac{1}{2}}` 👁 $\Big|\frac{1}{2}\Big|$ This is the main goal in Section 4 about syntactical commands for mathematics. A next more support/helping section about abbreviations and names is Section 5.

The first main part on general semantic commands is Section 7.

While all these are loaded by default. The next part, Section 8, introduces semantic commands for specific topics. These are given in separate sub-packages and can be loaded if you work in this area and want to use the commands.

The package should be loaded late, since it might overwrite a few commands, currently most prominently `\d` which is overwritten by `cleveref` in case `minted` is loaded. So for more flexibility, there is the alternative command `\dInt`.

2 Package Options

shortbb use shorter notations for the blackboard-bold math letters `\C`, `\K`, `\N`, `\Q`, `\R`, `\Z`

3 Required Packages

amssymb.sty defines mathematical symbol fonts

ifthen.sty facilitates the definition of conditional commands

ifxetex.sty provides a way to check if a document is being processed by XeTeX and company

mathtools.sty provides lots of improvements for math typesetting (includes `amsmath.sty`)

xifthen.sty extends `ifthen.sty` by adding new boolean conditions

xparse.sty provides a high-level interface to define new commands

xspace.sty adds space depending on context

4 Syntax

The `mathsemantics-syntax.sty` package provides mainly symbols and short commands, which can be used in semantic definitions for ease of notation. They usually are rather simple commands without too many parameters.

4.1 Letters

ba. . . bz	lower-case **b**old-face letters `\br`, `\bf` $\boldsymbol{r}, \boldsymbol{f}$
bA. . . bZ	upper-case **b**old-face letters `\bR`, `\bF` $\boldsymbol{R}, \boldsymbol{F}$
balpha. . . bomega	lower-case **b**old-face Greek letters `\balpha`, `\boldeta` $\boldsymbol{\alpha}, \boldsymbol{\eta}$ (the latter being an exception)
bAlpha. . . bOmega	upper-case **b**old-face Greek letters `\bGamma`, `\bDelta` $\boldsymbol{\Gamma}, \boldsymbol{\Delta}$
bnull	**b**old-face zero `\bnull` $\mathbf{0}$
bone	**b**old-face one `\bone` $\mathbf{1}$
cA. . . cZ	upper-case **c**alligraphic letters `\cM`, `\cN` $\mathcal{M}, \mathcal{N}$
fA. . . fZ	upper-case **f**raktur letters `\fM`, `\fN`, `\fX` $\mathfrak{M}, \mathfrak{N}, \mathfrak{X}$
sA. . . sZ	upper-case **s**cript letters `\sM`, `\sN`, `\sX` $\mathscr{M}, \mathscr{N}, \mathscr{X}$
va. . . vz	lower-case letters with a **v**ector accent `\va`, `\vb` $\vec{a}, \vec{b}$
vA. . . vZ	upper-case letters with a **v**ector accent `\vA`, `\vB` $\vec{A}, \vec{B}$
valpha. . . vomega	lower-case Greek letters with a **v**ector accent `\valpha`, `\vbeta` $\vec{\alpha}, \vec{\beta}$

vAlpha. . . vOmega upper-case Greek letters with a **v**ector accent `\vGamma`, `\vDelta` $\vec{\Gamma}, \vec{\Delta}$

vnull vector zero `\vnull` $\vec{0}$

vone vector one `\vone` $\vec{1}$

bbA,...,bbZ blackboard-bold uppercase letters

`\bbC`, `\bbK`, `\bbN`, `\bbQ`, `\bbR`, `\bbS`, `\bbZ` $\mathbb{C}, \mathbb{K}, \mathbb{N}, \mathbb{Q}, \mathbb{R}, \mathbb{S}, \mathbb{Z}$

use the package option `shortbb` to introduce
`\C`, `\K`, `\N`, `\Q`, `\R`, `\Z` $\mathbb{C}, \mathbb{K}, \mathbb{N}, \mathbb{Q}, \mathbb{R}, \mathbb{Z}$ if not already defined elsewhere (i. e. they are not redefined, only *provided*.

4.2 Syntax Helpers

enclspacing provides spacing after the opening and before the closing delimiters for `\enclose`. This is by default set to be empty.

enclose is a command which encloses some content in scaled delimiters. It is meant as a helper to facilitate the definition of other commands. Its syntax is `\enclose[#1]{#2}{#3}{#4}`. The first (optional) argument is used to scale the delimiters to the standard amsmath sizes.[1] The second and fourth arguments specify the opening and closing delimiters, respectively. The third argument is the content to be enclosed.

`\enclose{[}{\dfrac{1}{2}}{]}` $[\dfrac{1}{2}]$

`\enclose[Big][{\dfrac{1}{2}}]` $\Bigl[\dfrac{1}{2}\Bigr]$

`\enclose[auto]{[}{\dfrac{1}{2}}{]}` $\left[\dfrac{1}{2}\right]$

`\enclose[none]{[}{\dfrac{1}{2}}{]}` $\dfrac{1}{2}$

Note 1. none is merely meant for testing when having arguments in brackets whether it is useful to omit them. You can also deactivate the absolute value vertical lines this way, so *use this option with care.*

Note 2. This command should normally be used only in the definition of other commands. For instance, `\abs` is using it internally. See `\paren` for the

[1] `big`, `Big`, `bigg`, `Bigg` or `auto`, which uses `left` and `right` as well as none to easily deactivate brackets.

nicer command to use

enclspacingSet provides spacing before and after the center delimiter `\encloseSet`. This is by default set to `\,`.

encloseSet is a command which encloses some content in scaled delimiters. It is meant as a helper to facilitate the definition of other commands. Its syntax is `\encloseSet#1]{#2}{#3}{#4}{#5}{#6}`. The first (optional) argument is used to scale the delimiters including the center one to the standard amsmath sizes.[1] The second and sixth arguments specify the opening and closing delimiters, respectively. The fourth argument specifies the center delimiter and The third and fifth argument are the content to be enclosed.

`\encloseSet[big]{\{}{x\in\bbR}{|}{x>5}{\}}` $\big\{x \in \mathbb{R} \big| x > 5\big\}$

`\encloseSet[auto]{\{}{x\in\bbR}{|}{x>\dfrac{1}{2}}{\}}` $\left\{x \in \mathbb{R} \,\middle|\, x > \dfrac{1}{2}\right\}$

Note. This command should normally be used only in the definition of other commands. For instance, `\setDef` is using it internally.

paren is an alternative to `\enclose`, with a different ordering of arguments. Its syntax is `\paren[#1]{#2}{#3}{#4}`, which is simply mapped to `\enclose[#1]{#2}{#4}{#3}`.

`\paren[Big]{[}{]}{\dfrac{1}{2}}` $\Big[\dfrac{1}{2}\Big]$

`\paren[Big][]{\dfrac{1}{2}}` $\Big[\dfrac{1}{2}\Big]$

`\paren[auto]{[}{]}{\dfrac{1}{2}}` $\left[\dfrac{1}{2}\right]$

4.3 Spacing Helpers

clap complements the standard LaTeX commands `\llap` and `\rlap`. These commands horizontally smash their arguments.

`Let us \llap{smash} something.` Let us smash something.

`Let us \clap{smash} something.` Let us smash something.

`Let us \rlap{smash} something.` Let us smash something.

mathllap corresponds to `\llap` in math mode.

`\sum_{\mathllap{1\le i\le j\le n}} X_{ij}` $\sum_{1\le i\le j\le n} X_{ij}$

mathclap corresponds to `\clap` in math mode.

`\sum_{\mathclap{1\le i\le j\le n}} X_{ij}` $\sum_{1\le i\le j\le n} X_{ij}$

mathrlap corresponds to `\rlap` in math mode.

`\sum_{\mathrlap{1\le i\le j\le n}} X_{ij}` $\sum_{1\le i\le j\le n} X_{ij}$

mrep stands for *math replace* and it typesets an argument while reserving the space for another. Its syntax is `\mrep[#1]{#2}{#3}` The first (optional) argument is one of `{l,c,r}` and it is used to define the alignment. `c` is the default.

`\mrep[l]{1}{-1}-1` $1 \quad -1$

`\mrep[c]{1}{-1}-1` $\ 1 \ -1$

`\mrep[r]{1}{-1}-1` $\quad 1 - 1$

5 Abbreviations

5.1 English

aa almost all `\aa` a.a.

ale almost everwhere `\ale` a.e.

eg exempli gratia (for example) `\eg` e. g.

etc et cetera (and so on) `\etc` etc.

ie id est (id est) `\ie` i. e.

iid independent and identically distributed `\iid` i. i. d.

spd symmetric positive definite `\spd` s. p. d.

st such that or subject to `\st` s. t.

`wrt`	with respect to </> `\wrt` w.r.t.

5.2 German

`bspw`	beispielsweise (for example) </> `\bspw` bspw.
`bzgl`	bezüglich (with regard to) </> `\bzgl` bzgl.
`bzw`	beziehungsweise (respectively) </> `\bzw` bzw.
`Dah`	Das heißt (That is, beginning of phrase) </> `\Dah` D. h.
`dah`	das heißt (that is) </> `\dah` d. h.
`evtl`	eventuell (possibly) </> `\evtl` evtl.
`fs`	fast sicher </> `\fs` f. s.
`fue`	fast überall </> `\fue` f. ü.
`IA`	Im Allgemeinen (beginning of phrase) </> `\IA` I. A.
`iA`	im Allgemeinen </> `\iA` i. A.
`idR`	in der Regel </> `\idR` i. d. R.
`IdR`	In der Regel (beginning of phrase) </> `\IdR` I. d. R.
`iW`	im Wesentlichen </> `\iW` i. W.
`IW`	Im Wesentlichen (beginning of phrase) </> `\IW` I. W.
`mE`	meines Erachtens </> `\mE` m. E.
`oBdA`	ohne Beschränkung der Allgemeinheit </> `\oBdA` o. B. d. A.
`OBdA`	ohne Beschränkung der Allgemeinheit (beginning of phrase) </> `\OBdA` O. B. d. A.
`og`	oben genannt </> `\og` o. g.
`oae`	oder ähnliche </> `\oae` o. ä.

`so`	siehe oben `\so` s. t.
`ua`	unter anderem `\ua` u. a.
`Ua`	Unter anderem (beginning of phrase) `\Ua` U. a.
`ug`	unten genannt `\ug` u. g.
`usw`	und so weiter (and so on) usw.
`uU`	unter Umständen `\uU` u. U.
`UnU`	Unter Umständen (beginning of phrase) `\UnU` U. U.
`vgl`	vergleiche (compare) `\vgl` vgl.
`zB`	zum Beisiel `\zB` z. B.
`ZB`	Zum Beispiel (beginning of phrase) `\ZB` Z. B.
`zHd`	zu Händen `\zHd` z. Hd.

6 Names

`adimat`	ADiMat
`ampl`	AMPL
`BibTeX`	BibTeX
`BibLaTeX`	BibLaTeX
`cg`	CG
`cpp`	C++
`cppmat`	cppmat
`dolfin`	Dolfin
`dolfinplot`	Dolfin-Plot
`dolfinadjoint`	Dolfin-Adjoint

`doxygen`	Doxygen
`femorph`	FEMorph
`fenics`	FEniCS
`ffc`	FFC
`fmg`	FMG
`fortran`	Fortran
`gitlab`	GitLab
`gmres`	Gmres
`gmsh`	Gmsh
`ipopt`	Ipopt
`libsvm`	LIBSVM
`liblinear`	LIBLINEAR
`macmpec`	MacMPEC
`manifoldsjl`	Manifolds.jl
`manopt`	Manopt
`manoptjl`	Manopt.jl
`mathematica`	Mathematica
`matlab`	Matlab
`maple`	Maple
`maxima`	Maxima
`metis`	Metis
`minres`	Minres
`mshr`	mshr
`mvirt`	MVIRT

`numpy`	NumPy
`paraview`	ParaView
`pdflatex`	PDFLaTeX
`perl`	Perl
`petsc`	PETSc
`pymat`	pymat
`python`	Python
`scikit`	SciKit
`scikitlearn`	SciKit-learn
`scipy`	SciPy
`sphinx`	Sphinx
`subgmres`	SubGmres
`subminres`	SubMinres
`superlu`	SuperLU
`svmlight`	SVM$^{\text{light}}$
`tritetmesh`	TriTetMesh
`ufl`	UFL
`uqlab`	UQLab
`viper`	Viper
`xml`	XML

7 Semantic Commands

Build upon Syntax from Section 4 this part provides semantical mathematical commands.

abs absolute value. Its syntax is `\abs[#1]{#2}`. The first (optional) argument is used to scale the deliminters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the argument.

`\abs{a}` $|a|$

`\abs[Big]{\dfrac{1}{2}}` $\Big|\dfrac{1}{2}\Big|$

`\abs[auto]{\dfrac{1}{2}}` $\left|\dfrac{1}{2}\right|$

aff affine hull `\aff` aff

arcosh area hyperbolic cosine `\arcosh` arcosh

arcoth area hyperbolic cotangens `\arcoth` arcoth

argmax maximizer of a function `\argmax_{x \in \bbR} f(x)` $\operatorname*{arg\,max}_{x\in\mathbb{R}} f(x)$

Argmax set of maximizers of a function `\Argmax_{x \in \bbR} f(x)` $\operatorname*{Arg\,max}_{x\in\mathbb{R}} f(x)$

argmin minimizer of a function `\argmin_{x \in \bbR} f(x)` $\operatorname*{arg\,min}_{x\in\mathbb{R}} f(x)$

Argmin set of minimizers of a function `\Argmin_{x \in \bbR} f(x)` $\operatorname*{Arg\,min}_{x\in\mathbb{R}} f(x)$

arsinh area hyperbolic cotangens `\arsinh` arsinh

artanh area hyperbolic tangens `\artanh` artanh

bdiv bold (meaning: vector) divergence of a matrix-valued function `\bdiv` $\operatorname{\mathbf{div}}$

ceil integer larger or equal to input. Its syntax is `\ceil[#1]{#2}`. The first (optional) argument is used to scale the delimiters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the argument.

`\ceil{a}` $\lceil a\rceil$

`\ceil[Big]{\dfrac{1}{2}}` $\Big\lceil\dfrac{1}{2}\Big\rceil$

clconv closure of the convex hull of a set `\clconv M` $\overline{\operatorname{conv}}\, M$

closure closure of a set `\closure M` $\operatorname{cl} M$

cofac cofactor matrix `\cofac(A)` $\operatorname{cof}(A)$

compactly compact embedding of topological spaces `\compactly` $\hookrightarrow\hookrightarrow$

cone conic hull `\cone` cone

conv convex hull of a set `\conv M` $\operatorname{conv} M$

corresponds binary operator for correspondence `A\corresponds B` $A \mathrel{\widehat{=}} B$

cov covariance `\cov` Cov

curl the curl operator `\curl` curl

d, dInt integral symbol with prepended space, as in

`\int_\bbR \exp(-x^2) \d x` $\displaystyle\int_{\mathbb{R}} \exp(-x^2)\,\mathrm{d}x$

Since `\d` is often overridden, `\dInt` is the safe alternative

dev deviator of a matrix `\dev A` $\operatorname{dev} A$

diag diagonal matrix composed of entries in a vector, or diagonal of a matrix

`\diag(a)` $\operatorname{diag}(a)$

`\diag(A)` $\operatorname{diag}(A)$

diam diameter `\diam(M)` $\operatorname{diam}(M)$

distOp the mathematical operator denoting the distance

`\distOp` dist

dist distance from a point to a set. Its syntax is `\dist[#1]{#2}{#3}` or `\dist[#1]{#2}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set. The third argument denotes the point; it can be omitted. The command `\distOp` is used to typeset the operator.

`\dist[Big]{\cC}{\dfrac{x}{2}}` $\operatorname{dist}_{\mathcal{C}}\Big(\dfrac{x}{2}\Big)$

`\dist{\cC}` dist_C

`\dist` dist

`div` divergence `\div` div

`Div` (row-wise) divergence `\Div` Div

`dom` domain `\dom` dom

`dotcup` distinct union `\dotcup` $\mathbin{\dot\cup}$

`dprod` double contraction of matrices $A : B = \sum_{i,j} A_{ij} B_{ij} = \mathrm{trace}(A^\top B)$

`A \dprod B` $A : B$

`dual` duality pairing. Its syntax is `\dual[#1]{#2}{#3}`. The first (optional) argument is used to scale the delimiters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the first factor. The third argument denotes the second factor.

`\dual{x^*}{x}` $\langle x^*, x\rangle$

`\dual[Big]{x^*}{\dfrac{1}{2}}` $\Big\langle x^*, \dfrac{1}{2}\Big\rangle$

`e` Euler's number `\e` e

`embed` embedding of topological spaces `\embed` $\hookrightarrow$

`embeds` synonym of `\embed` `\embeds` $\hookrightarrow$

`epi` epigraph `\epi` epi

`eR` extended real line `\eR = \bbR \cup \{\pm \infty\}` $\overline{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$

`essinf` essential infimum

`\displaystyle\essinf_{x \in \bbR} f(x)` $\displaystyle\operatorname*{ess\,inf}_{x\in\mathbb{R}} f(x)$

`esssup` essential supremum

`\displaystyle\esssup_{x \in \bbR} f(x)` $\displaystyle\operatorname*{ess\,sup}_{x\in\mathbb{R}} f(x)$

`file` typesets a file name (using `nolinkurl`)

`\file{test.txt}` → `test.txt`

`floor` integer less or equal to input. Its syntax is `\floor[#1]{#2}`. The first (optional) argument is used to scale the delimiters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the argument.

`\floor{a}` → $\lfloor a \rfloor$

`\floor[Big]{\dfrac{1}{2}}` → $\Big\lfloor \dfrac{1}{2} \Big\rfloor$

`grad` gradient (of a function) `\grad F` → $\operatorname{grad} F$

`Graph` graph of a function `\Graph` → Graph

`id` identity operator `\id` → id

`image` image of a function `\image` → image

`inj` injedctivity (radius) `\inj` → inj

`inner` inner product. Its syntax is `\inner[#1]{#2}{#3}`. The first (optional) argument is used to scale the parentheses enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the first factor. The third argument denotes the second factor.

`\inner{a}{b}` → (a, b)

`\inner[Big]{a}{\dfrac{b}{2}}` → $\Big(a, \dfrac{b}{2}\Big)$

`interior` `\interior` → int

`jump` jump of a quantity, e. g., across a finite element facet. Its syntax is `\jump[#1]{#2}`. The first (optional) argument is used to scale the delimiters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the argument.

`\jump{a}` → $[\![a]\!]$

`\jump[Big]{\dfrac{1}{2}}` → $\Big[\!\!\Big[\dfrac{1}{2} \Big]\!\!\Big]$

`laplace` the Laplace operator `\laplace u` → Δu

`lin` linear hull of a set of vectors `\lin\{v_1,v_2\}` $\operatorname{lin}\{v_1, v_2\}$

`norm` norm of a vector. Its syntax is `\norm[#1]{#2}`. The first (optional) argument is used to scale the delimiters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the argument.

`\norm{a}` $\|a\|$

`\norm[Big]{\dfrac{c}{2}}` $\Big\|\dfrac{c}{2}\Big\|$

`\norm[auto]{\dfrac{c}{2}}` $\left\|\dfrac{c}{2}\right\|$

`projOp` the mathematical operator denoting the projection `\projOp` proj

`\projOp` proj

`proj` projection onto a set. Its syntax is `\proj[#1]{#2}(#3)` or `\proj[#1]{#2}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set and can also be left out. The third argument denotes the point; it can be omitted. The command `\projOp` is used to typeset the operator.

`\proj` proj

`\proj(x)` $\operatorname{proj}(x)$

`\proj{\cC}` proj_C

`\proj{\cC}(x)` $\operatorname{proj}_C(x)$

`\proj[Big](\dfrac{x}{2})` $\operatorname{proj}\Big(\dfrac{x}{2}\Big)$

`\proj[Big]{\cC}(\dfrac{x}{2})` $\operatorname{proj}_C\Big(\dfrac{x}{2}\Big)$

`proxOp` the mathematical operator denoting the proximal map

`\proxOp` prox

`prox` the proximal operator of a function. Its syntax is `\prox[#1]{#2}(#3)` or `\prox[#1]{#2}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set. The third argument denotes the point; it can be omitted. The command `\proxOp` is used to typeset the operator.

`\prox` prox

`\prox{\lambda F}` $\operatorname{prox}_{\lambda F}$

`\prox{\lambda F}(x)` $\operatorname{prox}_{\lambda F}(x)$

`\prox[auto]{\lambda F}(\dfrac{x}{2})` $\operatorname{prox}_{\lambda F}\left(\dfrac{x}{2}\right)$

rank rank (of a matrix) `\rank` rank

range range of some operator `\range` range

restr restriction/evaluation. Its syntax is `\restr[#1]{#2}{#3}`. The first (optional) argument is used to scale the deliminters enclosing the arguments to the standard amsmath sizes.[1] The second argument denotes the argument to be restricted/evaluated. The third argument denotes the restriction set/evaluation point.

`\restr[auto]{\frac{\d}{\d t}(f \circ \gamma)(t)}{t=0}` $\left.\frac{\mathrm{d}}{\mathrm{d}t}(f \circ \gamma)(t)\right|_{t=0}$

ri relative inerior `\ri` ri

setDef define a set, where `\setMid` serves as the center divider. Its syntax is `\setDef[#1]{#2}{#3}`. The first (optional) argument is used to scale the parantheses enclosing the argument and the center divider to the standard amsmath sizes.[1] The second argument denotes the left part of the definition, naming the potential elements of the set being defined. The third argument denotes the condition to include the elements in the set.

`\setDef{x\in\bbR}{x>5}` $\{x \in \mathbb{R} \,|\, x > 5\}$

`\setDef[Big]{x\in\bbR}{x>\dfrac{1}{2}}` $\Big\{x \in \mathbb{R} \,\Big|\, x > \dfrac{1}{2}\Big\}$

setMid divider within `\setDef` (set definitions).
This defaults to `\setMid` $|$.

sgn sign `\sgn` sgn

Sgn sign (set valued) `\Sgn` Sgn

supp support (of a function) `\supp F` $\operatorname{supp} F$

sym symmetric part (of a matrix) `\sym A` $\operatorname{sym} A$

`trace` trace (of a matrix) `\trace A` trace A

`transp` transpose of a vector or matrix.

`A^\transp` A^{T}

`transposeSymbol` symbol to use for the transpose

`\transposeSymbol` T

`var` variance `\var` Var

`weakly` weak convergence of a sequence `\weakly` $\rightharpoonup$

`weaklystar` weak star convergence of a sequence `\weaklystar` $\overset{\star}{\rightharpoonup}$

8 Additional Semantics by Topic

While semantic commands might be suitable for all mathematical topics, the following subsections collect commands which are most useful in one particular mathematical area and hence might clutter the general semantic file. Any semantic topic files should always build on `numapde-semantic.sty`.

8.1 Manifolds: `numapde-manifolds.sty`

The semantic file `numapde-manifolds.sty` collects definitions and notations for Riemannian manifolds.

`bitangentSpace` the bitangent space. Its syntax is `\bitangent{#1}[#2]`. The first argument denotes the base point. The second (optional) argument denotes the manifold, which defaults to $\mathcal{M}$.

`\bitangentSpace{p}` $\mathcal{T}_p^{**}\mathcal{M}$

`\bitangentSpace{q}[\cN]` $\mathcal{T}_q^{**}\mathcal{N}$

`bitangentSpaceSymbol` the symbol used within `\bitangentSpace`.

`\bitangentSpaceSymbol` $\mathcal{T}^{**}$

`cotangentSpace` the cotangent space. Its syntax is `\cotangentSpace{#1}[#2]`. The first argument denotes the base point. The second (optional) argument denotes the manifold, which defaults to $\mathcal{M}$.

`\cotangentSpace{p}` $\mathcal{T}_p^*\mathcal{M}$

`\cotangentSpace{q}[\cN]` $\mathcal{T}_q^*\mathcal{N}$

cotangentBundle the cotangent bundle. Its syntax is `\cotangentBundle[#1]`. The (optional) argument denotes the manifold, which defaults to $\mathcal{M}$.

`\cotangentBundle` $\mathcal{T}^*\mathcal{M}$

`\cotangentBundle[\cN]` $\mathcal{T}^*\mathcal{N}$

cotangentSpaceSymbol the symbol used within `\cotangent`.

`\cotangentSpaceSymbol` $\mathcal{T}^*$

covariantDerivative is the covariant derivative. Its syntax is `\covariantDerivative{#1}[#2]`. The first argument is the vector (or vector field) determining the direction of differentiation. The second (optional) argument denotes the tensor field being differentiated.

`\covariantDerivative{X}{Y}` $\mathrm{D}_X Y$

covariantDerivativeSymbol the symbol used for the covariant derivative `\covariantDerivative`.

`\covariantDerivativeSymbol` D

exponential the exponential map. Its syntax is `\exponential[#1]{#2}(#3)`. The first argument can be used to scale the third. The second argument denotes the base point and is mandatory. The third argument denotes the tangent vector, which is optional, but if provided, the argument is put in brackets. The first following example illustrates the case, where no brackets are put. Note that the space is mandatory.

`\exponential{p}X` $\exp_p X$

`\exponential{p}(X)` $\exp_p(X)$

`\exponential[Big]{p}(\frac{X}{2})` $\exp_p\Big(\frac{X}{2}\Big)$

expOp the symbol used within the `\exponential`.

`\expOp` $\exp$

geodesic

a geodesic. Its syntax is \geodesic#1–¡#2¿[#3]–#4˝–#5˝(#6)–.The first argument can be used to use a different symbol (locally) for the geodesic The second (optional) argument is used to modify the style of the geodesic (symbol, long, arc or plain, where the last is the default) The third (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] It is ignored when the sixth argument is not given. The fourth argument denotes the initial point (at $t = 0$). The fifth argument denotes either the final point (at $t = 1$) for types l and a, or the initial tangent vector for type p. The sixth (optional) argument denotes the evaluation point. The command \geodesicSymbol is used to typeset the geodesic symbol default (i.e. globally)

\geodesic<s> γ

\geodesic<s>(t) $\gamma(t)$

\geodesic<l>{p}{q} $\gamma(\cdot;p\,,\,q)$

\geodesic<l>{p}{q}(t) $\gamma(t;p\,,\,q)$

\geodesic<a>{p}{q} $\gamma_{\widehat{p,q}}$

\geodesic<a>[Big]{p}{q}(\dfrac{t}{2}) $\gamma_{\widehat{p,q}}\Bigl(\dfrac{t}{2}\Bigr)$

\geodesic<p>{p}{X} $\gamma_{p,X}$

\geodesic<p>{p}{X}(t) $\gamma_{p,X}(t)$

\geodesic<p>[Big]{p}{X}(\dfrac{t}{2}) $\gamma_{p,X}\Bigl(\dfrac{t}{2}\Bigr)$

\geodesic[big]{p}{X}((1-t)t) $\gamma_{p,X}\bigl((1-t)t\bigr)$

\geodesic|\dot\gamma|{p}{X}(t) $\dot\gamma_{p,X}(t)$

geodesicSymbol

symbol to use for the geodesic in \geodesic

\geodesicSymbol γ

inverseRetract

use an inverse retraction, the arguments are similar to \logarithm but use the \retractionSymbol

\inverseRetract{p}q $\operatorname{retr}_p^{-1}q$

\inverseRetract{p}(q) $\operatorname{retr}_p^{-1}(q)$

`\inverseRetract[Big]{p}(q)` $\operatorname{retr}_p^{-1}\Big(q\Big)$

logarithm the logarithmic map. Its syntax is `\logarithm[#1]{#2}(#3)`. The first argument can be used to scale the third. The second argument denotes the base point and is mandatory. The third argument denotes another point, which is optional, but if provided, the argument is put in brackets. The first following example illustrates the case, where no brackets are put. Note that the space is mandatory.

`\logarithm{p}q` $\log_p q$

`\logarithm{p}(q)` $\log_p(q)$

`\logarithm[Big]{p}(q)` $\log_p\Big(q\Big)$

logOp the symbol used within the `\logarithm`.

`\logOp` $\log$

parallelTransport the parallel transport.
Its syntax is `\parallelTransport[#1]{#2}{#3}(#4){5}`. The first (optional) argument is used to scale the parantheses enclosing the argument #4.[1] The second argument is the start point of parallel transport on a manifold. The third argument is the end point of parallel transport on a manifold. The fourth (optional) argument is the tangent vector that is transported. Putting it in brackets enables the scaling by the first argument. The fifth (optional) argument specifies an exponent, for example to parallel transport along a curve c

`\parallelTransport{p}{q}X` $\mathrm{P}_{q\gets p}X$

`\parallelTransport{p}{q}(X)` $\mathrm{P}_{q\gets p}(X)$

`\parallelTransport[big]{p}{q}(X)` $\mathrm{P}_{q\gets p}\big(X\big)$

`\parallelTransport{p}{q}(X)[c]` $\mathrm{P}_{q\gets p}^{c}(X)$

`\parallelTransport{p}{q}[c]` $\mathrm{P}_{q\gets p}^{c}$

parallelTransportDir similar to `\parallelTransport`, but the third argument is a direction to transport into. This can be rewritten to the classical notation applying an exponential map from the base point (#2) to th direction (#3). The fifth (optional) argument specifies an exponent, for example to parallel transport along a curve c

`\parallelTransportDir{p}{Y}X` $\mathrm{P}_{p,Y}X$

`\parallelTransportDir{p}{Y}(X)` $\mathrm{P}_{p,Y}(X)$

`\parallelTransportDir[big]{p}{Y}(X)` $\mathrm{P}_{p,Y}\big(X\big)$

`\parallelTransportDir{p}{Y}(X)[c]` $\mathrm{P}^{c}_{p,Y}(X)$

`\parallelTransportDir{p}{Y}[c]` $\mathrm{P}^{c}_{p,Y}$

parallelTransportSymbol the symbol to use within `\parallelTransport` and `\parallelTransportDir`

`\parallelTransportSymbol` P

retract a retraction.
Its syntax is `\retract[#1]{#2}{#3}`. The first argument can be used to scale the third. The second argument denotes the base point. The third argument denotes the tangent vector, which is optional, but if provided, the argument is put in brackets. The first following example illustrates the case, where no brackets are put. Note that the space is mandatory.

`\retract{p}X` retr_pX

`\retract{p}(X)` $\operatorname{retr}_p(X)$

`\retract[Big]{p}(\frac{X}{2})` $\operatorname{retr}_p\Big(\frac{X}{2}\Big)$

retractionSymbol symbol to use for a retraction and an inverse retraction, see `\retract` and `\inverseRetract`.

`\retractionSymbol` retr

riemannian the Riemannian metric (family of inner products on the tangent spaces). Its syntax is `\riemannian[#1]{#2}{#3}[#4]`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the first factor. The third argument denotes the second factor. The fourth (optional) argument denotes the base point of the tangent space.

`\riemannian{X_1}{X_2}` (X_1, X_2)

`\riemannian{Y_1}{Y_2}[q]` $(Y_1, Y_2)_q$

`\riemannian[Big]{\dfrac{1}{2}X_1}{X_2}[p]` $\Big(\dfrac{1}{2}X_1, X_2\Big)_p$

riemanniannorm the norm induced by the Riemannian metric.
Its syntax is `\riemanniannorm[#1]{#2}[#3]`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the argument. The third (optional) argument denotes the base point of the tangent space.

`\riemanniannorm{X}` $\|X\|$

`\riemanniannorm{Y}[p]` $\|Y\|_p$

`\riemanniannorm[Big]{\dfrac{1}{2}X}[p]` $\Big\|\dfrac{1}{2}X\Big\|_p$

secondCovariantDerivative is the second-order covariant derivative.
Its syntax is `\secondCovariantDerivative{#1}{#2}[#3]`. The first argument is the vector (or vector field) determining the first direction of differentiation. The second argument is the vector (or vector field) determining the second direction of differentiation. The third (optional) argument denotes the tensor field being differentiated.

`\secondCovariantDerivative{X}{Y}{T}` $\mathrm{D}^2_{X,Y}T$

secondCovariantDerivativeSymbol the symbol used for the second covariant derivative.
This is used within `\secondCovariantDerivative`.

`\secondCovariantDerivativeSymbol` D^2

tangentSpace the tangent space. Its syntax is `\tangentSpace{#1}[#2]`. The first argument denotes the base point. The second (optional) argument denotes the manifold, which defaults to $\mathcal{M}$.

`\tangentSpace{p}` $\mathcal{T}_p\mathcal{M}$

`\tangentSpace{q}[\cN]` $\mathcal{T}_q\mathcal{N}$

tangentBundle the tangent bundle. Its syntax is `\tangentBundle[#1]`. The (optional) argument denotes the manifold, which defaults to $\mathcal{M}$.

`\tangentBundle` $\mathcal{T}\mathcal{M}$

`\tangentBundle[\cN]` $\mathcal{T}\mathcal{N}$

tangentSpaceSymbol the symbol used within `\tangent`.

`\tangentSpaceSymbol` $\mathcal{T}$

tensorBundle

the tensor bundle. Its syntax is `\tensorBundle{#1}{#2}[#3]`. The first argument denotes the number r of elements of the cotangent space the tensors accept. The second argument denotes the number s of elements of the tangent space the tensors accept. The third (optional) argument denotes the manifold, which defaults to $\mathcal{M}$.

`\tensorBundle{r}{s}` $\mathcal{T}^{(r,s)}\mathcal{M}$

`\tensorBundle{r}{s}[\cN]` $\mathcal{T}^{(r,s)}\mathcal{N}$

tensorSpace

a tensor space over a vector space V. Its syntax is `\tensorSpace{#1}{#2}[#3]`. The first argument denotes the number r of elements of the dual space V^* the tensors accept. The second argument denotes the number s of elements of the space V the tensors accept. The third (optional) argument denotes the vector space, which defaults to empty.

`\tensorSpace{r}{s}` $\mathcal{T}^{(r,s)}()$

`\tensorSpace{r}{s}[V]` $\mathcal{T}^{(r,s)}(V)$

tensorSpaceSymbol

the symbol used within `\tensorSpace` and `\tensorBundle`.

`\tensorSpaceSymbol` $\mathcal{T}$

vectorTransport

a vector transport.
Its syntax is `\vectorTransport[#1]{#2}{#3}(#4)[#5]`. The first (optional) argument is used to scale the parantheses enclosing the argument #4.[1] The second argument is the start point of vector transport on a manifold. The third argument is the end point of vector transport on a manifold. The fourth (optional) argument is the tangent vector that is transported. Putting it in brackets enables the scaling by the first argument. Finally a retraction symbol can be added in the exponent to distinguish vector transports as #5.

`\vectorTransport{p}{q}X` $\mathrm{T}_{q\gets p}X$

`\vectorTransport{p}{q}(X)` $\mathrm{T}_{q\gets p}(X)$

`\vectorTransport[big]{p}{q}(X)` $\mathrm{T}_{q\gets p}\big(X\big)$

`\vectorTransport{p}{q}(X)[\retractionSymbol]` $\mathrm{T}^{\mathrm{retr}}_{q\gets p}(X)$

vectorTransportDir

similar to `\vectorTransport`, but the third argument is a direction to transport into. This can be rewritten to the classical notation applying an retraction from the base point (#2) to th direction (#3).

`\vectorTransportDir{p}{Y}X` $\mathrm{T}_{p,Y}X$

`\vectorTransportDir{p}{Y}(X)` $\mathrm{T}_{p,Y}(X)$

`\vectorTransportDir[big]{p}{Y}(X)` $\mathrm{T}_{p,Y}\big(X\big)$

`\vectorTransportDir{p}{Y}(X)[\retractionSymbol]` $\mathrm{T}^{\mathrm{retr}}_{p,Y}(X)$

vectorTransportSymbol the symbol to use within `\vectorTransport` and `\vectorTransportDir`

`\vectorTransportSymbol` T

8.2 Optimization: numapde-optimization.sty

The semantic file `mathsemantics-optimization.sty` collects definitions and notations related to optimization.

linearizingcone the linearizing cone. Its syntax is `\linearizingcone[#1]{#2}{#3}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set. The third argument denotes the base point.

`\linearizingcone{A}{x}` $\mathcal{T}_A^{\mathrm{lin}}(x)$

`\linearizingcone{A}{x^2}` $\mathcal{T}_A^{\mathrm{lin}}(x^2)$

`\linearizingcone[big]{A}{x^2}` $\mathcal{T}_A^{\mathrm{lin}}\big(x^2\big)$

normalcone the normal cone. Its syntax is `\normalcone[#1]{#2}{#3}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set. The third argument denotes the base point.

`\normalcone{A}{x}` $\mathcal{N}_A(x)$

`\normalcone{A}{x^2}` $\mathcal{N}_A(x^2)$

`\normalcone[big]{A}{x^2}` $\mathcal{N}_A\big(x^2\big)$

polarcone the polar cone of a set `\polarcone{A}` A°

radialcone the radial cone. Its syntax is `\radialcone[#1]{#2}{#3}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set. The third argument denotes the base point.

`\radialcone{A}{x}` $\mathcal{K}_A(x)$

`\radialcone{A}{x^2}` $\mathcal{K}_A(x^2)$

`\radialcone[big]{A}{x^2}` $\mathcal{K}_A\big(x^2\big)$

tangentcone the tangent cone. Its syntax is `\tangentcone[#1]{#2}{#3}`. The first (optional) argument is used to scale the parantheses enclosing the argument to the standard amsmath sizes.[1] The second argument denotes the set. The third argument denotes the base point.

`\tangentcone{A}{x}` $\mathcal{T}_A(x)$

`\tangentcone{A}{x^2}` $\mathcal{T}_A(x^2)$

`\tangentcone[big]{A}{x^2}` $\mathcal{T}_A\big(x^2\big)$