The mleftright package

Heiko Oberdiek
<heiko.oberdiek at googlemail.com>

2010/09/25 v1.0

Abstract

TEX sets subformulas by `\left` and `\right` as inner formulas with additional surrounding spaces in some situations. This package provides `\mleft` and `\mright` that call `\left` and `\right`, but the delimiters will act as normal `\mathopen` and `\mathclose` delimiters without the additional space of an inner formula.

Contents

1 Documentation **1**
1.1 Use . . . 2

2 Implementation **2**

3 Test **7**
3.1 Catcode checks for loading . . . 7

4 Installation **8**
4.1 Download . . . 8
4.2 Bundle installation . . . 9
4.3 Package installation . . . 9
4.4 Refresh file name databases . . . 9
4.5 Some details for the interested . . . 9

5 Catalogue **10**

6 Acknowledgement **10**

7 References **10**

8 History **11**
[2010/09/25 v1.0] . . . 11

9 Index **11**

1 Documentation

The package is a result of a thread in the newsgroup comp.text.tex with the subject *spacing after `\right)` and before `\left)`* [1]. The problem: `\left` and `\right` adjust the size of the delimiters automatically. However, TEX treats the whole expression as inner formula. In some circumstances TEX adds extra space before or after an inner formula. Example:

```
$\sin(x^2), x$                 ⇒  sin(x^2), x
$\sin\left(x^2\right), x$      ⇒  sin (x^2) , x
$\sin\mleft(x^2\mright), x$    ⇒  sin(x^2), x
```

(`\mleft` and `\mright` are provided by this package.)

In the newsgroup Donald Arseneau answered with clever macros [2]:

```
\newcommand\lft{\mathopen{}\left}
\newcommand\rgt{\aftergroup\mathclose\aftergroup{\aftergroup}\right}
```

However one problem remains, a following subscript or superscript is not applied to the right delimiter but the empty `\mathclose`. Thus Philipp Stephani provided an improvement [3]:

```
\mathopen{} \mathclose{\left\| A^2 \right\|}_2
```

Heiko Oberdiek converted this into macro form [4]:

```
\newcommand\lft{\mathopen{}\mathclose\bgroup\left}
\newcommand\rgt{\aftergroup\egroup\right}
```

The package uses longer macro names `\mleft` and `\mright` to avoid name clashes. Also it adds some checks for error conditions.

1.1 Use

`\mleft`⟨*delimL*⟩ ... `\mright`⟨*delimR*⟩

Macros `\mleft` and `\mright` are used in the same way as `\left` and `\right`. Also `\middle` can be used inbetween if ε-TEX is present.

`\mleftright`

Macro `\mleftright` redefines `\left` as `\mleft` and `\right` as `\mright`. The redefinition is local to the group.

`\mleftrightrestore`

Macro `\mleftright` restores `\left` and `\right` with the original meaning if they were previously redefined by `\mleftright` (also locally).

2 Implementation

```
1 ⟨*package⟩
```

Reload check, especially if the package is not used with LATEX.

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3   \catcode13=5 % ^^M
4   \endlinechar=13 %
5   \catcode35=6 % #
6   \catcode39=12 % '
7   \catcode44=12 % ,
8   \catcode45=12 % -
9   \catcode46=12 % .
10  \catcode58=12 % :
11  \catcode64=11 % @
12  \catcode123=1 % {
13  \catcode125=2 % }
14  \expandafter\let\expandafter\x\csname ver@mleftright.sty\endcsname
15  \ifx\x\relax % plain-TeX, first loading
```

```
  \else
    \def\empty{}%
    \ifx\x\empty % LaTeX, first loading,
      % variable is initialized, but \ProvidesPackage not yet seen
    \else
      \expandafter\ifx\csname PackageInfo\endcsname\relax
        \def\x#1#2{%
          \immediate\write-1{Package #1 Info: #2.}%
        }%
      \else
        \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
      \fi
      \x{mleftright}{The package is already loaded}%
      \aftergroup\endinput
    \fi
  \fi
\endgroup%
```

Package identification:

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode35=6 % #
  \catcode39=12 % '
  \catcode40=12 % (
  \catcode41=12 % )
  \catcode44=12 % ,
  \catcode45=12 % -
  \catcode46=12 % .
  \catcode47=12 % /
  \catcode58=12 % :
  \catcode64=11 % @
  \catcode91=12 % [
  \catcode93=12 % ]
  \catcode123=1 % {
  \catcode125=2 % }
  \expandafter\ifx\csname ProvidesPackage\endcsname\relax
    \def\x#1#2#3[#4]{\endgroup
      \immediate\write-1{Package: #3 #4}%
      \xdef#1{#4}%
    }%
  \else
    \def\x#1#2[#3]{\endgroup
      #2[{#3}]%
      \ifx#1\@undefined
        \xdef#1{#3}%
      \fi
      \ifx#1\relax
        \xdef#1{#3}%
      \fi
    }%
  \fi
\expandafter\x\csname ver@mleftright.sty\endcsname
\ProvidesPackage{mleftright}%
  [2010/09/25 v1.0 Math left/right delim. as open/close (HO)]%
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode123=1 % {
  \catcode125=2 % }
  \catcode64=11 % @
  \def\x{\endgroup
    \expandafter\edef\csname mleftright@AtEnd\endcsname{%
```

```
      \endlinechar=\the\endlinechar\relax
      \catcode13=\the\catcode13\relax
      \catcode32=\the\catcode32\relax
      \catcode35=\the\catcode35\relax
      \catcode61=\the\catcode61\relax
      \catcode64=\the\catcode64\relax
      \catcode123=\the\catcode123\relax
      \catcode125=\the\catcode125\relax
    }%
  }%
\x\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\def\TMP@EnsureCode#1#2{%
  \edef\mleftright@AtEnd{%
    \mleftright@AtEnd
    \catcode#1=\the\catcode#1\relax
  }%
  \catcode#1=#2\relax
}
\TMP@EnsureCode{38}{4}% &
\TMP@EnsureCode{39}{12}% '
\TMP@EnsureCode{40}{12}% (
\TMP@EnsureCode{41}{12}% )
\TMP@EnsureCode{42}{12}% *
\TMP@EnsureCode{43}{12}% +
\TMP@EnsureCode{44}{12}% ,
\TMP@EnsureCode{45}{12}% -
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{60}{12}% <
\TMP@EnsureCode{91}{12}% [
\TMP@EnsureCode{93}{12}% ]
\edef\mleftright@AtEnd{%
  \mleftright@AtEnd
  \escapechar\the\escapechar\relax
  \noexpand\endinput
}
\escapechar=92 %
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname RequirePackage\endcsname\relax
  \input infwarerr.sty\relax
  \input ltxcmds.sty\relax
\else
  \RequirePackage{infwarerr}[2010/04/08]%
  \RequirePackage{ltxcmds}[2010/04/26]%
\fi
```

The original commands `\left` and `\right` are saved and later used in `\mleft` and `\mright` in order to deal with:

```
\let\left\mleft
\let\right\mright
```

`\mleftright@OrgLeft`

```
\let\mleftright@OrgLeft\left
```

`\mleftright@OrgRight`

```
\let\mleftright@OrgRight\right
```

`\mleftright@Def` Macro `\mleftright@Def` defines a macro as robust macro if ε-TEX or LATEX is available.

```
\ltx@IfUndefined{protected}{%
  \ltx@IfUndefined{DeclareRobustCommand}{%
    \def\mleftright@Def{\def}%
  }{%
    \def\mleftright@Def{\DeclareRobustCommand*}%
  }%
}{%
  \def\mleftright@Def{\protected\def}%
}
\edef\mleftright@Def#1{%
  \noexpand\ltx@IfUndefined{%
    \noexpand\expandafter\noexpand\ltx@gobble\noexpand\string#1%
  }{%
    \expandafter\noexpand\mleftright@Def#1%
  }{%
    \noexpand\@PackageError{mleftright}{%
      Command \noexpand\string#1 already defined%
    }\noexpand\@ehd
    \noexpand\ltx@gobble
  }%
}
```

In case of ε-TEX the group status after the left symbol is saved and later checked at the beginning of `\mright`.

```
\ltx@IfUndefined{currentgrouplevel}{%
  \catcode38=14 % & = comment
}{%
  \catcode38=9 % & = ignore
}
```

`\mleftright@GroupLevel`

```
& \def\mleftright@GroupLevel{-1}%
```

`\mleftright@WrongGroup`

```
& \def\mleftright@WrongGroup#1(#2){%
&   \ifnum\mleftright@GroupLevel<\ltx@zero
&     \@PackageError{mleftright}{%
&       Missing previous \string\mleft
&     }\@ehc
&   \else
&     \@PackageError{mleftright}{%
&       Unexpected group status for \string\mright%
&       \ifnum\mleftright@GroupLevel=#1 %
&       \else
&         .\MessageBreak
&         Group level is #1, %
&           expected is \mleftright@GroupLevel
&       \fi
&       \ifnum16=#2 %
&       \else
&         .\MessageBreak
&         Group type is #2 (%
&         \ifcase#2 %
&           bottom level%
&           \expandafter\expandafter\expandafter\ltx@gobblefour
&           \expandafter\ltx@gobbletwo
&         \or simple%
&         \or hbox%
&         \or adjusted hbox%
&         \or vbox%
```

```
&         \or vtop%
&         \or align%
&         \or no align%
&         \or output%
&         \or math%
&         \or disc%
&         \or insert%
&         \or vcenter%
&         \or math choice%
&         \or semi simple%
&         \or math shift%
&         \or math left%
&         \else
&           unknown%
&         \fi
&         \space group),\MessageBreak
&         expected is 16 (math left group)%
&       \fi
&     }\@ehd
&   \fi
& }%
```

\mleft

```
\mleftright@Def\mleft{%
  \mathopen{}\mathclose\bgroup
& \edef\mleftright@GroupLevel{\the\numexpr\the\currentgrouplevel+1}%
  \mleftright@OrgLeft
}
```

\mright

```
\mleftright@Def\mright{%
& \ifnum\mleftright@GroupLevel=\currentgrouplevel
&   \ifnum16=\currentgrouptype
      \aftergroup\egroup
&   \else
&     \expandafter\mleftright@WrongGroup
&     \the\expandafter\currentgrouplevel
&     \expandafter(\the\currentgrouptype)%
&   \fi
& \else
&   \expandafter\mleftright@WrongGroup
&   \the\expandafter\currentgrouplevel
&   \expandafter(\the\currentgrouptype)%
& \fi
  \mleftright@OrgRight
}
```

\mleftright

```
\mleftright@Def\mleftright{%
  \let\left\mleft
  \let\right\mright
}
```

\mleftrightrestore

```
\mleftright@Def\mleftrightrestore{%
  \ifx\left\mleft
    \let\left\mleftright@OrgLeft
  \fi
  \ifx\right\mright
    \let\right\mleftright@OrgRight
  \fi
}
```

```
\mleftright@AtEnd%
⟨/package⟩
```

3 Test

3.1 Catcode checks for loading

```
⟨*test1⟩
\catcode`\{=1 %
\catcode`\}=2 %
\catcode`\#=6 %
\catcode`\@=11 %
\expandafter\ifx\csname count@\endcsname\relax
  \countdef\count@=255 %
\fi
\expandafter\ifx\csname @gobble\endcsname\relax
  \long\def\@gobble#1{}%
\fi
\expandafter\ifx\csname @firstofone\endcsname\relax
  \long\def\@firstofone#1{#1}%
\fi
\expandafter\ifx\csname loop\endcsname\relax
  \expandafter\@firstofone
\else
  \expandafter\@gobble
\fi
{%
  \def\loop#1\repeat{%
    \def\body{#1}%
    \iterate
  }%
  \def\iterate{%
    \body
      \let\next\iterate
    \else
      \let\next\relax
    \fi
    \next
  }%
  \let\repeat=\fi
}%
\def\RestoreCatcodes{}
\count@=0 %
\loop
  \edef\RestoreCatcodes{%
    \RestoreCatcodes
    \catcode\the\count@=\the\catcode\count@\relax
  }%
\ifnum\count@<255 %
  \advance\count@ 1 %
\repeat

\def\RangeCatcodeInvalid#1#2{%
  \count@=#1\relax
  \loop
    \catcode\count@=15 %
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\def\RangeCatcodeCheck#1#2#3{%
```

```
  \count@=#1\relax
  \loop
    \ifnum#3=\catcode\count@
    \else
      \errmessage{%
        Character \the\count@\space
        with wrong catcode \the\catcode\count@\space
        instead of \number#3%
      }%
    \fi
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\def\space{ }
\expandafter\ifx\csname LoadCommand\endcsname\relax
  \def\LoadCommand{\input mleftright.sty\relax}%
\fi
\def\Test{%
  \RangeCatcodeInvalid{0}{47}%
  \RangeCatcodeInvalid{58}{64}%
  \RangeCatcodeInvalid{91}{96}%
  \RangeCatcodeInvalid{123}{255}%
  \catcode`\@=12 %
  \catcode`\\=0 %
  \catcode`\%=14 %
  \LoadCommand
  \RangeCatcodeCheck{0}{36}{15}%
  \RangeCatcodeCheck{37}{37}{14}%
  \RangeCatcodeCheck{38}{47}{15}%
  \RangeCatcodeCheck{48}{57}{12}%
  \RangeCatcodeCheck{58}{63}{15}%
  \RangeCatcodeCheck{64}{64}{12}%
  \RangeCatcodeCheck{65}{90}{11}%
  \RangeCatcodeCheck{91}{91}{15}%
  \RangeCatcodeCheck{92}{92}{0}%
  \RangeCatcodeCheck{93}{96}{15}%
  \RangeCatcodeCheck{97}{122}{11}%
  \RangeCatcodeCheck{123}{255}{15}%
  \RestoreCatcodes
}
\Test
\csname @@end\endcsname
\end
```

⟨/test1⟩

4 Installation

4.1 Download

Package. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/oberdiek/mleftright.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/mleftright.pdf Documentation.

Bundle. All the packages of the bundle 'oberdiek' are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/oberdiek.tds.zip

[1] ftp://ftp.ctan.org/tex-archive/

TDS refers to the standard "A Directory Structure for TEX Files" (CTAN:tds/tds.pdf). Directories with texmf in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the oberdiek.tds.zip in the TDS tree (also known as texmf tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

Script installation. Check the directory TDS:scripts/oberdiek/ for scripts that need further installation steps. Package attachfile2 comes with the Perl script pdfatfi.pl that should be installed in such a way that it can be called as pdfatfi. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

4.3 Package installation

Unpacking. The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain TEX:

```
tex mleftright.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as texmf tree):

```
mleftright.sty            → tex/generic/oberdiek/mleftright.sty
mleftright.pdf            → doc/latex/oberdiek/mleftright.pdf
test/mleftright-test1.tex → doc/latex/oberdiek/test/mleftright-test1.tex
mleftright.dtx            → source/latex/oberdiek/mleftright.dtx
```

If you have a docstrip.cfg that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

4.4 Refresh file name databases

If your TEX distribution (teTEX, mikTEX, ...) relies on file name databases, you must refresh these. For example, teTEX users run texhash or mktexlsr.

4.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the .dtx source file. It can be extracted by AcrobatReader 6 or higher. Another option is pdftk, e.g. unpack the file into the current directory:

```
pdftk mleftright.pdf unpack_files output .
```

Unpacking with LATEX. The .dtx chooses its action depending on the format:

plain TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{mleftright.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLaTeX:

```
pdflatex mleftright.dtx
makeindex -s gind.ist mleftright.idx
pdflatex mleftright.dtx
makeindex -s gind.ist mleftright.idx
pdflatex mleftright.dtx
```

5 Catalogue

The following XML file can be used as source for the TeX Catalogue. The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `mleftright.xml`.

```
⟨*catalogue⟩
<?xml version='1.0' encoding='us-ascii'?>
<!DOCTYPE entry SYSTEM 'catalogue.dtd'>
<entry datestamp='$Date$' modifier='$Author$' id='mleftright'>
  <name>mleftright</name>
  <caption>Variants of delimiters that act as maths open/close.</caption>
  <authorref id='auth:oberdiek'/>
  <copyright owner='Heiko Oberdiek' year='2010'/>
  <license type='lppl1.3'/>
  <version number='1.0'/>
  <description>
    The package defines variants <tt>\mleft</tt> and <tt>\mright</tt>
    of <tt>\left</tt> and <tt>\right</tt>, that make the delimiters
    act as <tt>\mathopen</tt> and <tt>\mathclose</tt>.  These commands
    address spacing difficulties in subformulas.
    <p/>
    The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
  </description>
  <documentation details='Package documentation'
      href='ctan:/macros/latex/contrib/oberdiek/mleftright.pdf'/>
  <ctan file='true' path='/macros/latex/contrib/oberdiek/mleftright.dtx'/>
  <miktex location='oberdiek'/>
  <texlive location='oberdiek'/>
  <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
</entry>
⟨/catalogue⟩
```

6 Acknowledgement

Donald Arsenau: He provided the main trick and the first macros.

Philipp Stephani: He solved the subscript problem.

7 References

[1] Dave94705, *spacing after `\right)` and before `\left)`*, newsgroup comp.text.tex, Message-ID: `5d264909-7c3d-4c9d-9b22-434178b2bf90@g21g2000prn.googlegroups.com`, 2010-08-12.
http://groups.google.com/group/comp.text.tex/msg/e5b6833da7dc29bf

[2] Donald Arseneau, *Re: spacing after `\right)` and before `\left)`*, newsgroup comp.text.tex, Message-ID: `yfivd6svl8y.fsf@mutant.triumf.ca`, 2010-08-30.
http://groups.google.com/group/comp.text.tex/msg/e0b2e4386e5d04e4

[3] Philipp Stephani, *Re: spacing after `\right)` and before `\left)`*, newsgroup comp.text.tex, Message-ID: `4c8c8c1e$0$6981$9b4e6d93@newsspool4.arcor-online.net`, 2010-09-12.
http://groups.google.com/group/comp.text.tex/msg/87ac1f61321de3ef

[4] Heiko Oberdiek, *Re: spacing after `\right)` and before `\left)`*, newsgroup comp.text.tex, Message-ID: `i6jcc2$8of$1@news.eternal-september.org`, 2010-09-12.
http://groups.google.com/group/comp.text.tex/msg/257aa6119bef878b

8 History

[2010/09/25 v1.0]

- The first version.

9 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

Symbols

\# 242
\% 318
\@ 243, 316
\@PackageError 145, 159, 163
\@ehc 161
\@ehd 147, 201
\@firstofone 251, 254
\@gobble 248, 256
\@undefined 58
\\ 317
\{ 240
\} 241

A

\advance 281, 289, 304
\aftergroup 29, 212

B

\body 260, 264

C

\catcode 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 69, 70, 72, 73, 74, 78, 79, 80, 81, 82, 83, 84, 87, 88, 90, 91, 92, 93, 97, 99, 152, 154, 240, 241, 242, 243, 278, 287, 295, 299, 316, 317, 318
\count@ 245, 274, 278, 280, 281, 285, 287, 288, 289, 293, 295, 298, 299, 303, 304
\countdef 245
\csname 14, 21, 50, 66, 76, 121, 244, 247, 250, 253, 308, 335
\currentgrouplevel 206, 210, 215, 220
\currentgrouptype 211, 216, 221

D

\DeclareRobustCommand 134

E

\empty 17, 18
\end 336
\endcsname 14, 21, 50, 66, 76, 121, 244, 247, 250, 253, 308, 335
\endinput 29, 117
\endlinechar 4, 35, 71, 77, 89
\errmessage 297
\escapechar 116, 119

I

\ifcase 175
\ifnum 158, 165, 171, 210, 211, 280, 288, 295, 303
\ifx . . . 15, 18, 21, 50, 58, 61, 121, 230, 233, 244, 247, 250, 253, 308
\immediate 23, 52
\input 122, 123, 309
\iterate 261, 263, 265

L

\left 128, 226, 230, 231, 350
\LoadCommand 309, 319
\loop 259, 275, 286, 294
\ltx@gobble 141, 148
\ltx@gobblefour 177
\ltx@gobbletwo 178
\ltx@IfUndefined . . 130, 131, 140, 151
\ltx@zero 158

M

\mathclose 205, 351
\mathopen 205, 351
\MessageBreak 167, 173, 198
\mleft *2*, 160, 204, 226, 230, 349
\mleftright *2*, 225
\mleftright@AtEnd 95, 96, 114, 115, 237
\mleftright@Def 130, 204, 209, 225, 229
\mleftright@GroupLevel 156, 158, 165, 169, 206, 210
\mleftright@OrgLeft . . . 128, 207, 231
\mleftright@OrgRight . . 129, 223, 234
\mleftright@WrongGroup . 157, 214, 219
\mleftrightrestore *2*, 229
\mright 164, 209, 227, 233, 349

N

\next 265, 267, 269
\number 300
\numexpr 206

P

\PackageInfo 26
\protected 137
\ProvidesPackage 19, 67

R

\RangeCatcodeCheck 292, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331
\RangeCatcodeInvalid 284, 312, 313, 314, 315
\repeat 259, 271, 282, 290, 305
\RequirePackage 125, 126
\RestoreCatcodes . . 273, 276, 277, 332
\right 129, 227, 233, 234, 350

S

\space 198, 298, 299, 307

T

\Test 311, 334
\the 77, 78, 79, 80, 81, 82, 83, 84, 97, 116, 206, 215, 216, 220, 221, 278, 298, 299
\TMP@EnsureCode 94, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113

W

\write 23, 52

X

\x 14, 15, 18, 22, 26, 28, 51, 56, 66, 75, 87