`perfectcut.sty` documentation

Guillaume Munch-Maccagnoni*

3rd September 2017

Contents

1 Use

Original use This package supplies the following commands:

Command	Produces
`\perfectcut{#1}{#2}`	⟨#1 ‖ #2⟩
`\perfectbra{#1}`	⟨#1 |
`\perfectket{#1}`	| #1⟩

Command	Produces
`\perfectcase{#1}`	[#1|#2]
`\perfectbrackets{#1}`	[#1]
`\perfectparens{#1}`	(#1)

The effect of these commands is to let the delimiters grow according to the number of nested `\perfect`*`command`*s (regardless of the size of contents). The package is originally intended for solving a notational issue regarding the representation of abstract-machine-like calculi in proof theory and computer science.

*https://gitlab.com/gadmm/perfectcut

General use The package also defines `\perfectunary` and `\perfectbinary` for defining custom delimiters that behave similarly to the above ones. These commands should be considered when facing the need of nested delimiters that consistently grow regardless of the contents. (See "Advanced Use".)

2 Example

Using `perfectcut.sty`

```
\usepackage{perfectcut}
\let\cut\perfectcut
```

The following states the idempotency of an adjunction:

$$\left\langle t \,\middle\|\, \bar{\mu}x.\left\langle \mu\alpha.\langle u \parallel e\rangle \,\middle\|\, e'\right\rangle\right\rangle = \left\langle \mu\alpha.\left\langle t \,\middle\|\, \bar{\mu}x.\langle u \parallel e\rangle\right\rangle \,\middle\|\, e'\right\rangle$$

The following states the commutativity of a strong monad:

$$\left\langle t \,\middle\|\, \bar{\mu}x.\left\langle u \,\middle\|\, \bar{\mu}y.\langle v \parallel e\rangle\right\rangle\right\rangle = \left\langle u \,\middle\|\, \bar{\mu}y.\left\langle t \,\middle\|\, \bar{\mu}x.\langle v \parallel e\rangle\right\rangle\right\rangle$$

Using `\underline` to mark redexes:

$$\begin{aligned}
&\delta(V, x.y, x.y)\\
&= \mu\star.\left\langle V \,\middle\|\, \left[\bar{\mu}x.\underline{\langle y \parallel \star\rangle} \,\middle|\, \bar{\mu}x.\underline{\langle y \parallel \star\rangle}\right]\right\rangle\\
&= \mu\star.\left\langle V \,\middle\|\, \underline{\left[\bar{\mu}x.\left\langle \iota_1(x) \,\middle\|\, \bar{\mu}z.\langle y \parallel \star\rangle\right\rangle \,\middle|\, \bar{\mu}x.\left\langle \iota_2(x) \,\middle\|\, \bar{\mu}z.\langle y \parallel \star\rangle\right\rangle\right]}\right\rangle\\
&= \mu\star.\left\langle V \,\middle\|\, \underline{\bar{\mu}z.}\langle y \parallel \star\rangle\right\rangle\\
&= \mu\star.\langle y \parallel \star\rangle = y
\end{aligned}$$

Using `\left`, `\middle` and `\right` instead

```
\renewcommand{\cut}[2]{\left\langle #1\middle|\mkern-2mu\middle|#2\right\rangle}
```

The following states the idempotency of an adjunction:

$$\left\langle t \middle\| \bar{\mu}x.\left\langle \mu\alpha.\langle u \| e\rangle \middle\| e' \right\rangle \right\rangle = \left\langle \mu\alpha.\langle t \| \bar{\mu}x.\langle u \| e\rangle\rangle \middle\| e' \right\rangle$$

The following states the commutativity of a strong monad:

$$\langle t \| \bar{\mu}x.\langle u \| \bar{\mu}y.\langle v \| e\rangle\rangle\rangle = \langle u \| \bar{\mu}y.\langle t \| \bar{\mu}x.\langle v \| e\rangle\rangle\rangle$$

Using `\underline` to mark redexes:

$$\begin{aligned}
&\delta(V, x.y, x.y)\\
&= \mu\star.\left\langle V \middle\| \left[\bar{\mu}x.\underline{\langle y \| \star\rangle} \middle| \bar{\mu}x.\underline{\langle y \| \star\rangle}\right] \right\rangle\\
&= \mu\star.\left\langle V \middle\| \underline{\left[\bar{\mu}x.\langle \iota_1(x) \| \bar{\mu}z.\langle y \| \star\rangle\rangle \middle| \bar{\mu}x.\langle \iota_2(x) \| \bar{\mu}z.\langle y \| \star\rangle\rangle\right]} \right\rangle\\
&= \mu\star.\left\langle V \middle\| \underline{\bar{\mu}z.}\,\langle y \| \star\rangle \right\rangle\\
&= \mu\star.\langle y \| \star\rangle = y
\end{aligned}$$

As we can see, the legibility of the above rendering is hampered by multiple issues: the delimiters grow inconsistently, vertical bars have the wrong size, accents or underlines uselessly make the delimiters grow, and the spacing could be improved. The package is designed to fix these issues.

3 Advanced use

The package lets you define your own growing delimiters. Let us first stress that the size of these delimiters is entirely determined by the number of nestings and is insensitive to the size of the contents. If you need the size of the contents to be taken into account then it is probably sufficient to use `\left` and `\right` while tweaking `\delimitershortfall` and `\delimiterfactor`.

Example

The following displays a set {#1 | #2} with delimiters appropriately sized if there are other `\perfectcommands` inside `#1` and `#2`.

```
\def\Set#1#2{\perfectbinary{IncreaseHeight}\{|\}{#1\mathrel{}}{\mathrel{}#2}}
\[\Set{\perfectparens{a}}{\perfectparens{b}}\]
```

$$\boxed{\left\{(a) \;\middle|\; (b)\right\}}$$

Custom delimiters

\perfectunary#1#2#3#4 Displays #2 #4 #3 where #2 and #3 are delimiters. The delimiters grow according to #1 which must be one of IncreaseHeight, CurrentHeight, or CurrentHeightPlusOne.

\perfectbinary#1#2#3#4#5#6 Displays #2 #5 #3 #6 #4 where #2, #3 and #4 are delimiters. The delimiters grow according to #1 which must be one of IncreaseHeight, CurrentHeight, or CurrentHeightPlusOne.

Stock delimiters

The stock commands behave as follow:

Command	Produces	Growth	Inserts skips
\perfectcut{#1}{#2}	⟨#1 ‖ #2⟩	IncreaseHeight	Yes
\perfectbra{#1}	⟨#1 |	IncreaseHeight	Yes
\perfectket{#1}	| #1⟩	IncreaseHeight	Yes
\perfectcase{#1}	[#1|#2]	CurrentHeightPlusOne	Yes
\perfectbrackets{#1}	[#1]	CurrentHeightPlusOne	Only inside
\perfectparens{#1}	(#1)	CurrentHeight	Only inside
\perfectunary{#1} {#2}{#3}{#4}	#2 #4 #3	#1	No
\perfectbinary{#1} {#2}{#3}{#4}{#5}{#6}	#2 #5 #3 #6 #4	#1	No

4 Options

Option mathstyle

With this option, the command \currentmathstyle from the package mathstyle is used instead of the command \ThisStyle from the scalerel package. The latter (default) uses \mathchoice, and in order to avoid an exponential time complexity in the number of nestings, it is only called for the outermost command. Concretely, with mathstyle, math style changes are taken into account both inside and outside of \perfectcommands, whereas without it, changes in math style are not obeyed inside a command. Moreover, using the mathstyle option speeds up compilation. The downside of using mathstyle is that it redefines many standard math commands, and is therefore a source of incompatibilities, notably with xypic.

Option realVert

With the option realVert, the double bars are obtained with the \Vert command. But without it, two \vert symbols are used and their spacing is controlled with \cutinterbarskip. Also, if

`realVert` is not activated, then a penalty (`binoppenalty`) is added, such that $\left\langle \mu\alpha.\langle a \parallel b\rangle \,\middle\|\, \bar{\mu}x.\langle c \parallel d\rangle \right\rangle$ splits across lines.

Option `fixxits`

For some reason that the author was unable to identify, the vertical alignment is wrong with the Opentype XITS math font with XeTeX. The option `fixxits` fixes this behaviour.

Customisation

The following mu-skips can be redefined in your preamble:

Command	Defines the spacing...
`\cutbarskip=5.0mu plus 8mu minus 2.0mu`	around bars
`\cutangleskip=0.0mu plus 8mu minus 1.0mu`	around delimiters (inside)
`\cutangleouterskip=0.0mu plus 8mu minus 0mu`	around delimiters (outside)
`\cutinterbarskip=0.8mu plus 0mu minus 0mu`	between bars (excl. `realVert`)

(1 mu equals 1/18 of an em in the current math font.)

5 Reimplementation of fixed-size delimiters

In addition, I provide the following corrections and generalisations of `\big`,`\bigg`, etc. Why not using the latter? Because both the plain TEX and the `amsmath` versions can be incorrect when changing the math font, the font size, the math style or `\delimitershortfall`. Moreover, Opentype math fonts sometimes offer more than five sizes. For this package we need a robust solution.

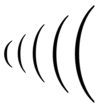
Command	Example	
#1-th size of delimiter #2		
`\nthleft{#1}{#2}`	`\nthleft{2}(`	$($
`\nthmiddle{#1}{#2}`	`\nthmiddle{2}\Vert`	$\Vert$
`\nthright{#1}{#2}`	`\nthright{2})`	$)$
delimiter #2 of height at least #1		
`\lenleft{#1}{#2}`	`\lenleft{3.2mm}[`	$[$
`\lenmiddle{#1}{#2}`	`\lenmiddle{3.2mm}|`	$\|$
`\lenright{#1}{#2}`	`\lenright{3.2mm}]`	$]$
delimiter #2 of height exactly #1 obtained by scaling the above one		
`\reallenleft{#1}{#2}`	`\reallenleft{3.2mm}[`	$[$
`\reallenmiddle{#1}{#2}`	`\reallenmiddle{3.2mm}|`	$\|$
`\reallenright{#1}{#2}`	`\reallenright{3.2mm}]`	$]$

Example with \nthleft

```
\nrthleft0(\nthleft1(\nthleft2(\nthleft3(\nthleft4(\nthleft5(\nthleft6(
```

Example with \big,\Big,\bigg,\Bigg

```
(\big(\Big(\bigg(\Bigg(
```



The above uses the \big commands from the amsmath package. The amsmath package corrects issues with the original TEX commands, but I could still notice inconsistencies, such as \big starting at size 2, under some font combinations. \nthleft, \nthright and \nthmiddle are implemented in a more robust way.

6 License

This work may be distributed and/or modified under the conditions of the LATEX Project Public License, either version 1.3 of this license or (at your option) any later version. Refer to the README file.