1 Introduction

The package mathpartir provides macros for displaying formulas and lists of formulas that are typeset in mixed horizontal and vertical modes.

The environment \texttt{mathpar} generalizes the math display mode to allow several formulas on the same line, and several lines in the same display. The arrangement of the sequence of formulas into lines is automatic depending on the line width and on a minimum inter-formula space alike words are displayed in a paragraphs (in centerline mode). A typical application is displaying a set of type inference rules.

The macro \texttt{inferrule} typesets inference rules. Both premises and conclusions are presented as lists of formulas and are typeset in paragraph mode and wrapped into multiple lines whenever necessary.

2 The \texttt{mathpar} environment

The mathpar environment is a “paragraph mode for formulas”. It allows to typeset long list of formulas putting as many as possible on the same line:

\begin{mathpar}
A\text{-Formula} \text{ \textand } \\
Longer\text{-Formula} \text{ \textand } \\
And \text{ \textand } The\text{-Last\text{-One}}
\end{mathpar}

Formulas are separated by \texttt{\textand} (or equivalently by a blank line). To enforce a vertical break it suffices to replace \texttt{\textand} by \texttt{\textbackslash}.
The implementation of \texttt{mathpar} entirely relies on the paragraph mode for text. It starts a new paragraph, and a math formula within a paragraph, after adjusting the spacing and penalties for breaks. Then, it simply binds \texttt{\&} to something like \texttt{\goodbreak}.

3 The \texttt{inferrule} macro

The \texttt{inferrule} macro is designed to typeset inference rules. It should only\footnote{Even though the basic version may work in text mode, we discourage its use in text mode; the star-version cannot be used in text mode} be used in math mode (or display math mode).

The basic use of the rule is

\begin{verbatim}
\texttt{\inferrule}
\{one \ \ two \ \ three \ \ or \ \ more \ \ premises\}
\{and \ \ any \ \ number \ \ of \ \ conclusions \ \ as \ \ well\}
\end{verbatim}

This is the rendering on a large page

\begin{tabular}{llll}
one & two & three & or more premises \\
and & any & number & of conclusions as well
\end{tabular}

However, the same formula on a narrower page will automatically be typeset like that:

\begin{tabular}{llllll}
one & two & three & or more premises \\
and & any & number & of conclusions as well
\end{tabular}

An inference rule is mainly composed of a premise and a conclusion. The premise and the conclusions are both list of formulas where the elements are separated by \\

Note the asymmetry between typesetting of the premises and of conclusions where lines closer to the center are fit first.

A newline can be forced by adding an empty line \\

\begin{verbatim}
\texttt{\inferrule}
\{aa \ \ \ \ \ \ \ bb\}
\{dd \ \ ee \ \ \ \ \ ff\}
\end{verbatim}

\begin{tabular}{llll}
aa & bb \\
dd & ee & ff
\end{tabular}
3.1 Single rules

Single rules are the default mode. Rules are aligned on their fraction bar, as illustrated below:

\[
\frac{aa}{bb} \quad \frac{bb}{ee}
\]

If the premise or the conclusion is empty, then the fraction bar is not typeset and the premise or the conclusion is centered:

\[
\inferrule{}{aa} + \quad \inferrule{aa}{aa + aa}
\]

Use use \{ \} instead of {} to get an axiom for instance:

\[
\inferrule{\{}{aa} + \quad \inferrule{aa}{aa + aa}
\]

The macro \texttt{\inferrule} accepts a label as optional argument, which will be typeset on the top left corner of the rule:

\[
\inferrule[yop]{aa \ \ bb}{cc}
\]

See section 3.6 for changing typesetting of labels. A label can also be placed next to the rule directly, since the rule is centered:

\[
\inferrule{aa \ \ bb}{cc} \quad (\textsc{Yop})
\]

3.2 Customizing presentation

By default, lines are centered in inference rules. However, this can be changed by either \texttt{\mprset{flushleft}} or \texttt{\mprset{center}}. For instance,

\[
\\$\$\texttt{\mprset{flushleft}} \quad \inferrule {a \ \ bb \ \ccc \ \ddd \ \ e \ \ ff \ \gg}
\]

Note that lines are aligned independently in the premise and the conclusion, which are both themselves centered. In particular, left alignment will not affect a single-line premise or conclusion.
3.3 Customizing rules

One may wish to change use rules for rewriting rule or implications, etc. There is a generic way of definition new rules by providing three parts: a tail, a body, and a head. The rule will then be built by joining all three components in this order and filling the body with leaders to extend as much as necessary. Here are examples

\[
\mprset{fraction={===}}
\inferrule{a \ \ bbb}{cc}
\]

\[
\mprset{fraction={\models=\Rightarrow}}
\inferrule{a \ \ bbb}{cc}
\]

The height and depth of the body are used to adjust vertical space. One, may “smash” the body to reduce the vertical space

\[
\mprset{fraction={\scriptstyle \vdash \smash-\rightarrow}}
\inferrule{a \ \ bbb}{cc}
\]

Since vertical skip does not take header and footer into account, which is usually better but sometimes odd, this can be adjusted explicitly:

\[
\mprset{fraction={|=/}, fractionaboveskip=0.6ex, fractionbelowskip=0.4ex}
\inferrule{a \ \ bbb_\downarrow}{cc^T\uparrow}
\]

Finally, it is also possible to provide its own definition of fraction by

\[
\def\over{\hbox{$#1 \over #2$}}
\mprset{myfraction=\over}
\inferrule{a \ \ bbb}{cc}
\]
Customizing the horizontal skip between premises (default value is 2em).

\$$\mprset{sep=6em}\infrule{a \ \ bbb}{cc}\$$

\[ a \quad bbb \]

\[ cc \]

Customizing the vertical space between premises (default value is empty). Notice that leaving it empty and setting vskip to 0em is not quite equivalent as show below between the third and fourth rules (because the typesetting cannot use the primitive typesetting of fractions).

\$$\def\R{\infrule{aa \ aa \ bbb \ bbb}{cc}\hspace{3em}}\R\mprset{vskip=0ex}\R\mprset{vskip=1ex}\R$$

\[ aa \quad aa \]
\[ bbb \quad bbb \]
\[ cc \]

\[ aa \quad aa \]
\[ bbb \quad bbb \]
\[ cc \]

\[ aa \quad aa \]
\[ bbb \quad bbb \]
\[ cc \]

\[ cc \]

3.4 Tabulars in inference rules

Although you probably do not want to do that, you may still use tabular or minipages inside inference rules, but between braces, as follows:

\infer [Tabular-Rule]
{some \ \ math \ \ and \ \ }
{\begin{tabular}{|l|r|}
\hline
Ugly & and \\
\hline
table & text \\
\hline
\end{tabular}}
{\begin{minipage}{6em}Do you really wish to do that?\end{minipage}}
{some \ \ conclusions}

\begin{tabular}{|l|}
\hline
Ugly & and \\
\hline
table & text \\
\hline
\end{tabular}

\text{Do you really wish to do that?}

\begin{tabular}{|l|}
\hline
some & conclusions \\
\hline
\end{tabular}
3.5 Derivation trees

To help writing cascades of rules forming a derivation tree, inference rules can also be aligned on their bottom line. For this, we use the star-version:

\begin{align*}
&\text{\texttt{\textbackslash inferrule*}} \\
&\{\text{\texttt{\textbackslash inferrule*}} \{\texttt{aa} \ \texttt{bb}\}\{\texttt{cc}\} \\
&\texttt{dd}\} \\
&\{\texttt{ee}\}
\end{align*}

The star version can also take an optional argument, but with a different semantics. The optional argument is parsed by the \texttt{keyval} package, so as to offer a set of record-like options:

<table>
<thead>
<tr>
<th>key</th>
<th>arg</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>before</td>
<td>\texttt{tex}</td>
<td>Execute \texttt{tex} before typesetting the rule. Useful for instance to change the maximal width of the rule.</td>
</tr>
<tr>
<td>width</td>
<td>\texttt{d}</td>
<td>Set the width of the rule to \texttt{d}</td>
</tr>
<tr>
<td>narrower</td>
<td>\texttt{d}</td>
<td>Set the width of the rule to \texttt{d} times \texttt{\hsize}.</td>
</tr>
<tr>
<td>lab</td>
<td>\texttt{ℓ}</td>
<td>Put label \texttt{ℓ} on the top of the rule as with the non-start version.</td>
</tr>
<tr>
<td>Lab</td>
<td>\texttt{ℓ}</td>
<td>same as lab</td>
</tr>
<tr>
<td>left</td>
<td>\texttt{ℓ}</td>
<td>Put label \texttt{ℓ} on the left of the rule</td>
</tr>
<tr>
<td>Left</td>
<td>\texttt{ℓ}</td>
<td>Idem, but as if label \texttt{ℓ} had zero width.</td>
</tr>
<tr>
<td>Right</td>
<td>\texttt{ℓ}</td>
<td>As \texttt{Left}, but on the right of the rule.</td>
</tr>
<tr>
<td>right</td>
<td>\texttt{ℓ}</td>
<td>As \texttt{left}, but on the right of the rule.</td>
</tr>
<tr>
<td>leftskip</td>
<td>\texttt{d}</td>
<td>Cheat by (skip negative space) \texttt{d} on the left side.</td>
</tr>
<tr>
<td>rightskip</td>
<td>\texttt{d}</td>
<td>Cheat by \texttt{d} on the right side of the rule.</td>
</tr>
<tr>
<td>vdots</td>
<td>\texttt{d}</td>
<td>Raise the rule by \texttt{d} and insert vertical dots.</td>
</tr>
</tbody>
</table>

We remind at the end the global options that we’ve seen above that can also
be set locally in derivation trees:

<table>
<thead>
<tr>
<th>sep</th>
<th>d</th>
<th>Set the separation between premises and conclusions to s.</th>
</tr>
</thead>
<tbody>
<tr>
<td>flushleft</td>
<td></td>
<td>flush premises to the left hand side</td>
</tr>
<tr>
<td>center</td>
<td></td>
<td>center premises on each line.</td>
</tr>
<tr>
<td>rewrite</td>
<td>d</td>
<td></td>
</tr>
<tr>
<td>myfraction</td>
<td>tex</td>
<td>set fraction to tex command</td>
</tr>
<tr>
<td>fraction</td>
<td>lmr</td>
<td>set fraction pattern to lmr with leaders.</td>
</tr>
<tr>
<td>vskip</td>
<td>d</td>
<td>Set the vertical skip between premises and conclusions to h.</td>
</tr>
<tr>
<td>vcenter</td>
<td></td>
<td>Make the rule centered around the fraction line as the non-star version</td>
</tr>
</tbody>
</table>

Here is an example of a complex derivation:

\[
\begin{array}{c}
  a \\
  \hline
  b b & c c & d d \\
  \hline
  e e
\end{array}
\quad
\begin{array}{c}
  \hline
  f f & g g \\
  \hline
  h h
\end{array}
\quad
\begin{array}{c}
  X X \\
  \hline
  u u & v v \\
  \hline
  w w
\end{array}
\]

(1) (when \(n > 0\))

and its code

\begin{verbatim}
\inferrule* [left=Total]
  {\inferrule*[Left=Foo]
    {\inferrule*[Right=Bar, rightstyle=\bf, leftskip=2em, rightskip=2em, vdots=1.5em]
      {a \ \ a \ \ \ \ bb \ \ cc \ \ dd}
      {ee}
    \ \ ff \ \ gg}
  \ \ hh}
\inferrule* [lab=XX]{uu \ \ vv}{ww})
\end{verbatim}

3.6 Label styles

The package uses \texttt{\DefTirNameStyle}, \texttt{\LabTirNameStyle}, \texttt{\LeftTirNameStyle}, and \texttt{\RightTirNameStyle} to typeset labels introduced with the default option,
Lab-, Left-, or Right-, respectively (or their uncapitalized variants). This can safely be redefined by the user. `\DefTirName` is normally used for defining occurrences (`i.e. in rule `\inferrule`) while the three other forms are used for referencing names (`i.e. in the star-version`). The styles can also be redefined using labeled-arguments of the star-version of `\inferrule` as described in table below.

Instead of just changing the style, the whole typesetting of labels may be changed by redefining `\DefTirName`, `\LabTirName`, `\LeftTirName`, and `\RightTirName`, each of which receives the label to be typeset as argument.

Finally, the vertical skip

<table>
<thead>
<tr>
<th>key</th>
<th>arg</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>style</td>
<td>tex</td>
<td>set the default style for labels to tex</td>
</tr>
<tr>
<td>leftstyle</td>
<td>tex</td>
<td>idem for labels</td>
</tr>
<tr>
<td>rightstyle</td>
<td>tex</td>
<td>idem for right labels</td>
</tr>
</tbody>
</table>

3.7 Star v.s. non-star version

The package also defines `\infer` as a shortcut for `\inferrule` but only if it is not previously defined.

There are two differences between the plain and star versions of `\inferrule`. The plain version centers the rule on the fraction line, while the star one centers the rule on the last conclusion, so as to be used in derivation trees.

Another difference is that the optional argument of the plain version is a label to always be placed on top of the rule, while the ∗-version takes a record of arguments. Hence, it can be parameterized in many more ways.

One may recover the plain version from the star version by passing the extra argument `vcenter` as illustrated below (the base line is aligned with the dotted line):

\[
\begin{array}{cc}
  aaaa \\
  aa & bb \\
  cc & cc \\
  \cdots & \cdots \\
  dd & dd
\end{array}
\]

\[
\begin{array}{cc}
  aaaa \\
  aa & bb \\
  cc & cc \\
  \cdots & \cdots \\
  dd & dd
\end{array}
\]

This is convenient, for instance to typeset rules with side conditions and keep them attached to the rule:

\[
\begin{array}{c}
\text{Pos} \\
\frac{aa}{cc} \frac{aa}{cc} (\text{if } n > 0)
\end{array}
\]

\[
\begin{array}{c}
\text{Neg} \\
\frac{aa}{cc} \frac{aa}{cc} (\text{if } n < 0)
\end{array}
\]
Or differently,

\[
\begin{array}{c}
\text{Pos} \\
\text{(if } n > 0) \\
\text{aaa} \\
\text{aaa} \\
\text{cc} \\
\hline
\text{Neg} \\
\text{(if } n < 0) \\
\text{aaa} \\
\text{aaa} \\
\text{cc}
\end{array}
\]

### 3.8 Implementation

The main macro in the implementation of inference rules is the one that either premises and conclusions. The macros uses two box-registers one \texttt{hbox} for typesetting each line and one \texttt{vbox} for collecting lines. The premise appears as a list with \ backslash as separator. Each element is considered in turn typeset in a \texttt{hbox} in display math mode. Its width is compare to the space left on the current line. If the box would not fit, the current horizontal line is transferred to the vertical box and emptied. Then, the current formula can safely be added to the horizontal line (if it does not fit, nothing can be done). When moved to the vertical list, lines are aligned on their center (as if their left-part was a left overlapped). At the end the vbox is readjusted on the right.

This description works for conclusions. For premises, the elements must be processes in reverse order and the vertical list is simply built upside down.

### 4 Other Options for the \texttt{mathpar} environment

The vertical space in \texttt{mathpar} is adjusted by \texttt{\MathparLineskip}. To restore the normal paragraph parameters in mathpar mode (for instance for some inner paragraph), use the command \texttt{\MathparNormalpar}. The environment uses \texttt{\MathparBindings} to rebind \ and \par. You can redefine thus command to change the default bindings or add your own.

### 5 Examples

See the source of this documentation —the file \texttt{mathpartir.tex}— for full examples.

### 6 H\textsc{E}V\textsc{E}A compatibility

The package also redefines \texttt{\hva} to do nothing in \texttt{mathpar} environment and in inference rules.

In H\textsc{E}V\textsc{E}A, \texttt{\textbackslash and} will always produce a vertical break in mathpar environment; to obtain a horizontal break, use \texttt{\hva \textbackslash and} instead. Conversely, \texttt{\textbackslash} will always produce a horizontal break in type inference rules; to obtain a vertical break, use \texttt{\hva \textbackslash} instead.

For instance, by default the following code,
\begin{mathpar}
\inferrule* \[ \text{Left=Foo} \]
{\inferrule* \[ \text{Right=Bar, width=8em,}
\quad \text{leftskip=2em, rightskip=2em, vdots=1.5em} \]
{a \quad a \quad bb \quad cc \quad dd}
{ee}
{ff \quad gg}
{hh}
\and
\inferrule* \[ \text{lab=XX} \]
{uu \quad vv}
{ww}
\end{mathpar}

which typesets in \TeX{} as follows,

\[
\begin{array}{cccc}
  & a & a & \quad bb & cc & dd \\
\hline
  ee & & & & & \\
  ff & & & & & \quad gg \\
\hline
  hh & & & & & \\
\end{array}
\]

\begin{array}{cccc}
  & a & a & \quad bb & cc & dd \\
\hline
  ee & & & & & \quad BAR \\
  ff & & & & & \quad ff & gg \\
\hline
  hh & & & & & \quad xx & uu & vv \quad ww \\
\end{array}
\]

would appear as follows with the compatible \texttt{HVPE}A mode:

\[
\begin{array}{cccc}
  & a & a & \quad bb & cc & dd \\
\hline
  ee & & & & & \quad BAR \\
  ff & & & & & \quad gg \\
\hline
  hh & & & & & \quad xx & uu & vv \quad ww \\
\end{array}
\]

To obtain (almost) the same rendering as in \TeX{}, it could be typed as

\begin{mathpar}
\inferrule* \[ \text{Left=Foo} \]
{\inferrule* \[ \text{Right=Bar, width=8em,}
\quad \text{leftskip=2em, rightskip=2em, vdots=1.5em} \]
{a \quad a \quad hva \quad bb \quad cc \quad dd}
{ee}
{ff \quad gg}
{hh}
\and
\hva \quad \text{and}
\inferrule* \[ \text{lab=XX} \]
{uu \quad vv}
{ww}
\end{mathpar}
Actually, it would be typeset and follows with the compatible H_{E}V_{E}A mode:

\[
\begin{array}{c}
\text{bb} & \text{cc} & \text{dd} \\
\text{ee} & \text{ee} & \text{ee} \\
\text{hh} & \text{ff} & \text{gg}
\end{array}
\]

XX

uu vv

ww

7 Implementation

1 \%
2 \%
3 \%
4 \%
5 \%
6 \%
7 \%
8 \%
9 \%
10 \%
11 \%
12 \%
13 \%
14 \%
15 \%
16 \%
17 \%
18 \%
19 \%
20 \%
21 \%
22 \%
23 \%
24 \%
25 \%
26 \%
27 \%
28 \%
29 \%
30 \%
31 \%
32 \%
33 \%
34 \%
35 \%
36 \%
37 \%
38 \%
39 \%

\text{PART I: Typesetting maths in paragraphe mode}

\text{\newdimen \mpr@tmpdim}

\text{Dimens are a precious ressource. Uses seems to be local.}

\text{To ensure hevea \hva compatibility, \hva should expands to nothing}

\text{in mathpar or in inferrule}

\text{normal paragraph parametters, should rather be taken dynamically}

\text{\def \mpr@savepar {%
\edef \MathparNormalpar {\noexpand \lineskiplimit \the\lineskiplimit
\noexpand \lineskip \the\lineskip} %
}}

\text{\def \mpr@rulelineskip {\lineskiplimit=0.3em \lineskip=0.2em plus 0.1em}}

\text{\def \mpr@lesslineskip {\lineskiplimit=0.6em \lineskip=0.5em plus 0.2em}}

\text{\def \mpr@lineskip {\lineskiplimit=1.2em \lineskip=1.2em plus 0.2em}}

\text{\let \MathparLineskip \mpr@lineskip}

\text{\def \mpr@paroptions {\MathparLineskip}}

\text{\let \mpr@prebindings \relax}

\text{\newskip \mpr@andskip \mpr@andskip 2em plus 0.5fil minus 0.5em}

\text{\def \mpr@goodbreakand
{\hskip \mpr@andskip \penalty -1000 \hskip \mpr@andskip}}

\text{\def \mpr@and {\hskip \mpr@andskip \penalty 50 \mpr@and}}

\text{\def \mpr@andcr {\penalty -10000 \mpr@and}}

11
\newtoks \mpr@lista
\newtoks \mpr@listb
\long \def \mpr@cons #1\mpr@to#2{\mpr@lista {\{#1}}\mpr@listb \expandafter {#2}\edef #2{\the \mpr@lista \the \mpr@listb}}
\long \def \mpr@snoc #1\mpr@to#2{\mpr@lista {\{#1}}\mpr@listb \expandafter {#2}\edef #2{\the \mpr@listb \the \mpr@lista}}
\long \def \mpr@concat#1=#2\mpr@to#3{\mpr@lista \expandafter {#2}\mpr@listb \expandafter {#3}\edef #1{\the \mpr@lista \the \mpr@listb}}
\def \mpr@head #1\mpr@to #2{\expandafter \mpr@head@ #1\mpr@head@ #1#2}
\long \def \mpr@head@ #1#2\mpr@head@ #3#4{\def #4{#1}\def #3{#2}}
\def \mpr@flatten #1\mpr@to #2{\expandafter \mpr@flatten@ #1\mpr@flatten@ #1#2\expandafter \mpr@stripof \mpr@all \mpr@stripped}
\loop \mpr@flatten \mpr@all \mpr@to \mpr@one \expandafter \mpr@snoc \mpr@one \mpr@to #2\expandafter \mpr@stripof \mpr@all \mpr@stripped \mpr@isempty 0 \else \let \mpr@isempty 1 \fi
\ifx 1\mpr@isempty \repeat
\def \mpr@makelist #1\mpr@to #2{\def \mpr@all {#1}\expandafter \mpr@lista \expandafter {\{\empty}\mpr@listb \expandafter {#2}\edef \mpr@all {\the \mpr@lista \the \mpr@listb \the \mpr@lista}}
\def \mpr@rev #1\mpr@to #2{\let \mpr@tmp \empty \def \##1{\mpr@cons \##1\mpr@to \mpr@tmp}#1\let #2\mpr@tmp}
\newif \if@premisse
\newbox \mpr@hlist
\newbox \mpr@vlist
\newif \ifmpr@center \mpr@centertrue
\def \mpr@vskip {}\def \mpr@htovlist {\setbox \mpr@hlist \hbox {\strut \mpr@center \hskip -0.5\wd \mpr@hlist} fi
\unhbox \mpr@hlist %
\def \mpr@makelist #1\mpr@to #2{\def \mpr@all {#1}
\expandafter \mpr@lista \expandafter {\{\empty}\mpr@listb \expandafter {#2}\edef \mpr@all {\the \mpr@lista \the \mpr@listb \the \mpr@lista}}
\def \mpr@rev #1\mpr@to #2{\let \mpr@tmp \empty \def \##1{\mpr@cons \##1\mpr@to \mpr@tmp}#1\let #2\mpr@tmp}
\newif \if@premisse
\if@premisse \newbox \mpr@hlist
\newbox \mpr@vlist
\newif \ifmpr@center \mpr@centertrue
\def \mpr@makelist #1\mpr@to #2{\def \mpr@all {#1} %
\expandafter \mpr@lista \expandafter {\{\empty}\mpr@listb \expandafter {#2}\edef \mpr@all {\the \mpr@lista \the \mpr@listb \the \mpr@lista}}
\def \mpr@rev #1\mpr@to #2{\let \mpr@tmp \empty
\def \##1{\mpr@cons \##1\mpr@to \mpr@tmp}#1\let #2\mpr@tmp}
%%% INFERENCE RULES

\def \mpr@fraction #1#2{\hbox {\advance \hsize by \hsize
\hbox {\mpr@over #2} \mpr@fraction #1#2}}

%%% A generic solution to arrow
\def \mpr@fractionaboveskip {0ex}
\def \mpr@fractionbelowskip {0.22ex}
\def \mpr@make@fraction #1#2#3#4#5{\hbox {% 
\def \mpr@tail{#1} 
\def \mpr@body{#2} 
\def \mpr@head{#3} 
\setbox1=\hbox{$#4$} \setbox2=\hbox{$#5$} 
\setbox3=\hbox{$\mkern -3mu \mpr@body \mkern -3mu$} 
\dimen0\ht3 \advance \dimen0 by \dp3 \relax 
\dimen0 0.5\dimen0 \relax 
\advance \dimen0 by \mpr@fractionaboveskip \dimen0 \relax 
\setbox2=\hbox {\raise \dimen0 \box2} \relax 
\setbox1=\hbox {\lower \dimen0 \box1} \relax 
\setbox0=\hbox {$\displaystyle {\box1 \atop \box2}$} 
\dimen0=\wd0 \box0 
\box0 \hskip -\dimen0 \relax 
\hbox to \dimen0 {$\%\color{blue} \mathrel{\mpr@tail} \joinrel 
\xleaders\hbox{\copy3}\hfil \joinrel \mathrel{\mpr@head} \%$} 
}}

%% Old stuff should be removed in next version
\def \mpr@nothing #1#2{$\lower 0.01pt \mpr@nofraction {#1}{#2}$}
\def \mpr@reduce #1#2{\hbox {\mpr@fraction {#1}{#2} \mkern -15mu \rightarrow}}
\def \mpr@rewrite #1#2#3{\hbox {\mpr@fraction {#2}{#3} \mkern -8mu #1}}
\def \mpr@infercenter #1{\vcenter {\mpr@hovbox{T}{#1}}}
\def \mpr@empty {} 
\def \mpr@inferrule \bgroup 
\ifnum \linewidth<\hsize \hsize \linewidth\fi 
\mpr@rulelineskip 
\let \and \quadquad 
\let \hva \mpr@hva 
\let \@rulename \mpr@empty 
\let \@rule@options \mpr@empty 
\let \mpr@over \@@over 
\mpr@inferrule\f]
\newcommand \mpr@inferrule@[3][] \{everymath={\displaystyle}\} 
\def \@test {#2/\text{ifixempty} \@test 
\setbox0=\hbox {$\vcenter {\mpr@hovbox{T}{#3}}$} \% 
\else 
\def \@test {#3/\text{ifixempty} \@test 

16
They are two forms
\begin{align}
\text{\inferrule [label][premisses]{conclusions}} \\
\text{\inferrule* [options][premisses]{conclusions}}
\end{align}

Premisses and conclusions are lists of elements separated by \\.
Each \\ produces a break, attempting horizontal breaks if possible, and vertical breaks if needed.
An empty element obtained by \\\ produces a vertical break in all cases.
The former rule is aligned on the fraction bar.
The optional label appears on top of the rule.
The second form to be used in a derivation tree is aligned on the last line of its conclusion.
The second form can be parameterized, using the key=val interface. The following keys are recognized:
\begin{itemize}
\item width
\item narrower
\item before
\item lab
\item lskip
\item left
\item Left
\item right
\item Right
\item leftskip
\item rightskip
\item vdots
\item after
\end{itemize}
Note that most options must come in this order to avoid strange typesetting (in particular, leftskip must precede left and Left and rightskip must follow Right or right; vdots must come last or be only followed by rightskip.)
% Keys that make sense in all kinds of rules
\def \mprset #1{\setkeys{mprset}{#1}}
\define@key {mprset}{andskip}[]{\mpr@andskip=#1}
\define@key {mprset}{lineskip}[]{\lineskip=#1}
\define@key {mprset}{lessskip}[]{\lineskip=0.5\lineskip}
\define@key {mprset}{flushleft}[]{\mpr@centerfalse}
\define@key {mprset}{center}[]{\mpr@centertrue}
\define@key {mprset}{rewrite}[]{\let \mpr@fraction \mpr@@rewrite}
\define@key {mprset}{atop}[]{\let \mpr@fraction \mpr@@nofraction}
\define@key {mprset}{fraction}[]{\def \mpr@fraction {\mpr@make@fraction #1}}
\define@key {mprset}{defaultfraction}[]{\let \mpr@fraction \mpr@@fraction}
\define@key {mprset}{sep}{\def \mpr@sep {#1}}
\define@key {mprset}{fractionaboveskip}{\def \mpr@@fractionaboveskip {#1}}
\define@key {mprset}{fractionbelowskip}{\def \mpr@@fractionbelowskip {#1}}
\define@key {mprset}{style}[1]{\def \TirNameStyle {#1}}
\define@key {mprset}{rightstyle}[1]{\def \RightTirNameStyle {#1}}
\define@key {mprset}{leftstyle}[1]{\def \LeftTirNameStyle {#1}}
\define@key {mprset}{vskip}[1]{\def \mpr@vskip {#1}}
\define@key {mpr}{flushleft}[]{\mpr@centerfalse}
\define@key {mpr}{center}[]{\mpr@centertrue}
\define@key {mpr}{rewrite}[]{\let \mpr@fraction \mpr@@rewrite}
\define@key {mpr}{myfraction}[]{\let \mpr@fraction {#1}}
\define@key {mpr}{fraction}[]{\def \mpr@fraction {\mpr@make@fraction #1}}
\define@key {mpr}{width}{\hsize #1}
\define@key {mpr}{sep}{\def \mpr@sep {#1}}
\define@key {mpr}{before}{#1}
\define@key {mpr}{lab}{\let \DefTirName \LabTirName \def \mpr@rulename {#1}}
\define@key {mpr}{Lab}{\let \DefTirName \LabTirName \def \mpr@rulename {#1}}
\define@key {mpr}{style}[1]{\def \TirNameStyle {#1}}
\define@key {mpr}{rightstyle}[1]{\def \RightTirNameStyle {#1}}
\define@key {mpr}{leftstyle}[1]{\def \LeftTirNameStyle {#1}}
\define@key {mpr}{vskip}[1]{\def \mpr@vskip {#1}}
\define@key {mpr}{narrower}{\hsize #1\hsize}
\define@key {mpr}{leftskip}{\hskip -#1}
\define@key {mpr}{reduce}[]{\let \mpr@fraction \mpr@@reduce}
\define@key {mpr}{rightskip}
\setbox \mpr@right \hbox {\unhbox \mpr@right \hskip -#1}
\define@key {mpr}{reduce}[]{\let \mpr@fraction \mpr@@reduce}
\define@key {mpr}{rightskip}
\setbox \mpr@right \hbox {\unhbox \mpr@right \hskip -#1}
\define@key {mpr}{left}{\setbox0 \hbox {$\LeftTirName {#1}\;$}\relax
\advance \hsize by -\wd0\box0}
\define@key {mpr}{left}{\setbox0 \hbox {$\LeftTirName {#1}\;$}\relax
\advance \hsize by -\wd0\box0}
\define@key {mpr}{right}
\def \TirNameStyle #1{\small \textsc{#1}}
\def \LeftTirNameStyle #1{\TirNameStyle {#1}}
\def \RightTirNameStyle #1{\TirNameStyle {#1}}
\def \lefttir@name #1{\hbox {\small \LeftTirNameStyle{#1}}}
\def \righttir@name #1{\hbox {\small \RightTirNameStyle{#1}}}
\let \TirName \lefttir@name
\let \LeftTirName \lefttir@name
\let \DefTirName \lefttir@name
\let \LabTirName \lefttir@name
\let \RightTirName \righttir@name

%%% Other Exports
\let \listcons \mpr@cons
\let \listsnoc \mpr@snoc
\let \listhead \mpr@head
\let \listmake \mpr@makelist