The \texttt{bussproofs-extra} package\textsuperscript{*}

Richard Zach

1 Introduction

The \texttt{bussproofs-extra} package provides additional functionality for the proof tree typesetting package \texttt{bussproofs} by Sam Buss. It is experimental and tested only with v.1.1, and only in \LaTeX{} mode with upward-growing trees. Functionality provided includes:

1. $\texttt{\textbackslash Deduce}$ and $\texttt{\textbackslash DeduceC}$ commands, which work much like $\texttt{\textbackslash Infer}$ commands but indicate missing parts of a proof.

2. Multiple styles for typesetting the result of $\texttt{\textbackslash Deduce}$, including
   (a) $\texttt{\textbackslash straightDeduce}$, which produces vertical dots
   (b) $\texttt{\textbackslash branchDeduce}$, which produces diagonal plus vertical dots
   (c) $\texttt{\textbackslash ddotsDeduce}$, which produces diagonal dots from top left to bottom right
   (d) $\texttt{\textbackslash dotsdDeduce}$, which produces diagonal dots from top right to bottom left

$\texttt{\textbackslash straightDeduce}$ is the default. It can be changed by redefining $\texttt{\textbackslash alwaysDeduce}$.

3. $\texttt{\textbackslash LeftLineLabel}$ and $\texttt{\textbackslash RightLineLabel}$ commands which work like $\texttt{\textbackslash LeftLabel}$ and $\texttt{\textbackslash RightLabel}$ but place a label next to the conclusion of an inference/deduction instead of the score line. \textbf{These don't work properly and may be removed}!

Here's what these deductions look like:

\textsuperscript{*}This document corresponds to \texttt{bussproofs-extra 0.3}, dated 2019/04/04.
The most up-to-date version of this package is available at the Open Logic Project github site, where you can file bug reports as well.

1.1 Example

```
\begin{prooftree}
\AxiomC{}
\RightLabel{$\pi_1(a)$}
\Deduce$\Gamma_1 \fCenter \Theta_1, F(a)$
\RightLabel{$\forall$R}
\UnaryInf$\Gamma_1 \fCenter \Theta_1, \forall x\,F(x)$
\ddotsDeduce
\RightLabel{$\pi_1'$}
\Deduce$\Gamma \fCenter \Theta, \forall x\,F(x)$
\AxiomC{}
\RightLabel{$\pi_2$}
\Deduce$F(n), \Delta_1 \fCenter \Lambda_1$
\RightLabel{$\forall$L}
\UnaryInf$\forall x\,F(x), \Delta_1 \fCenter \Lambda_1$
\dotsdDeduce
\RightLabel{$\pi_2'$}
\Deduce$\forall x\,F(x), \Delta \fCenter \Lambda$
\RightLabel{cut}
\BinaryInf$\Gamma, \Delta \fCenter \Theta, \Lambda$
\RightLabel{$\pi_4$}
\branchDeduce
\Deduce$\Pi \fCenter \Xi$
\end{prooftree}
```

produces this:
It is also possible to label entire subproofs on the left and on the right.

\begin{proof}
\AxiomC{}
\Deduce$\Gamma \fCenter \Delta$
\Deduce$\Gamma \fCenter \Delta, A$
\LeftSubproofLabel{$\pi$}
\AxiomC{}
\Deduce$\Gamma' \fCenter \Delta'$
\Deduce$A, \Gamma' \fCenter \Delta'$
\RightSubproofLabel{$\pi'$}
\RightLabel{cut}
\BinaryInf$\Gamma, \Gamma' \fCenter \Delta, \Delta'$
\Deduce$\Pi \fCenter \Lambda$
\end{proof}

2 Implementation

2.1 Setup

We require bussproofs (obviously) and tikz for drawing things.
2.2 Dimensions

`bussproofs` aligns sequents at the right end of the sequent arrow, so we need to remember by how much to correct to get deductions to the middle of sequents. For `\ddotsDeduce` and `\dotsdDeduce` (diagonal) styles, the upper and lower sequents will be displaced.

\newdimen\CenterCorrection
\newdimen\DiagCorrection

2.3 Deduce Styles

The following commands set the style for the next \`Deduce` command. \`\straightDeduce` produces a simple vertical line of dots, \`\branchDeduce` produces centered branching (Takeuti/Gentzen-style) dots, `\ddotsDeduce` left-to-right diagonal dots, and `\dotsdDeduce` right-to-left diagonal dots. They do this by redefining the \`fDeduce` command which produces the dots and sets up the dimensions. The `\alwaysDeduce` command is used to (re)set the deduce style to a default and is executed every time a deduction is typeset. It can be redefined to change the default deduce style.

\def\straightDeduce{%
  \gdef\fDeduce{\tikz\draw[very thick,loosely dotted] (0,0) -- (0,1);}
  \global\DiagCorrection=0pt
  \ignorespaces}

\def\branchDeduce{%
  \gdef\fDeduce{\begin{tikzpicture}
    \draw[very thick,loosely dotted] (0,0) -- (0,1);
    \draw[very thick,loosely dotted] (-.5,.5) -- (0,0);
    \draw[very thick,loosely dotted] (.5,.5) -- (0,0);
  \end{tikzpicture}}
  \global\DiagCorrection=0pt
  \ignorespaces}

\def\ddotsDeduce{%
  \gdef\fDeduce{\begin{tikzpicture}
    \draw[very thick,loosely dotted] (0,1) -- (1,0);
  \end{tikzpicture}}
  \setbox\myBoxA=\hbox{\fDeduce}
  \global\DiagCorrection=-\wd\myBoxA
  \ignorespaces}

\def\alwaysDeduce{%
  \gdef\fDeduce{\begin{tikzpicture}
    \draw[very thick,loosely dotted] (0,0) -- (1,0);
  \end{tikzpicture}}
  \global\DiagCorrection=0pt
  \ignorespaces}
2.4 \Deduce$ and \DeduceC

\Deduce$ and \DeduceC are the commands to actually produce the deductions. They are used and work just like \UnaryInf$ and \UnaryInfC.

\begin{verbatim}
def\Deduce$#1\fCenter#2${
  \prepUnary%
  \buildConclusion{#1}{#2}%
  \setbox\myBoxA=\hbox{\fCenter}
  \global\CenterCorrection=-.5\wd\myBoxA
  \joinDeduce%
  \resetInferenceDefaults%
  \ignorespaces%
}
def\DeduceC#1{
  \prepUnary%
  \buildConclusionC{#1}%
  \global\CenterCorrection=-4pt
  \joinDeduce%
  \resetInferenceDefaults%
  \ignorespaces%
}
\end{verbatim}

3 Typesetting the Deduction

\joinDeduce aligns and joins \curBox and \myBoxC into a single vbox. \curBox holds the upper proof, \curScoreStart is distance to where the line below the premise would start, \curScoreCenter is distance from left edge of score to the alignment point, and \curScoreEnd is width of the score line.
\global\advance\curCenter by -\hypKernAmt%

If center of premise is left of center of conclusion move upper box to right by difference, else move lower box right by difference

\ifnum\curCenter<\newCenter%
\displace=\newCenter%
\advance \displace by -\curCenter%
\kernUpperBox%
\else%
\displace=\curCenter%
\advance \displace by -\newCenter%
\kernLowerBox%
\fi%

For \ddotsDeduce, move lower box right; for \dotsdDeduce, move upper box right; then set \curCenter to align with horizontal center of dots.

\ifnum\DiagCorrection<0%
\displace=-\DiagCorrection
\kernLowerBox%
\else
\displace=\DiagCorrection
\kernUpperBox%
\fi%
\advance\curCenter by-.5\DiagCorrection

\ifnum\newScoreStart < \curScoreStart % \global \curScoreStart = \newScoreStart \fi%
% \ifnum \curScoreEnd < \newScoreEnd % \global \curScoreEnd = \newScoreEnd \fi%
% Leave room for the left label.
% \ifnum \curScoreStart<\wd\myBoxLL%
% \global\displace = \wd\myBoxLL%
% \global\advance\displace by -\curScoreStart%
% \kernUpperBox%
% \kernLowerBox%
% \fi%

Now we draw the deduction.
\buildDeduce%

Put the deduction and labels into a box.
\buildScoreLabels%

Put everything into a new box and compute the dimensions for the next \Deduce or \XxxxInf.
\ifx\rootAtBottomFlag\myTrue%
\buildRootBottom%
\else%
\buildRootTop%
\fi%
\global \curScoreStart=\newScoreStart%
\global \curScoreEnd=\newScoreEnd%
\buildDeduce does for \DeduceX what \buildInf does for \XxxInf: put the deduction bit (dots) into a box and set the dimensions properly.

\def\buildDeduce{\global\setbox \myBoxD = \hbox{\fDeduce}\displace = \wd\myBoxD \% find width of vdots

set start and end of current score to left and right of the box holding the deduction.

\global\curScoreStart = \curCenter
\global\advance\curScoreStart by -.5\displace
\global\curScoreEnd = \curCenter
\global\advance\curScoreEnd by .5\displace
\global\advance\curScoreStart by \CenterCorrection
\global\advance\curScoreEnd by \CenterCorrection
}

3.1 Line Labels

\LeftLineLabel and \RightLineLabel set the label to place to the left or right, respectively, of the conclusion of the next \Axiom, \XxxInf or \Deduce command. They are aligned with the text produced by \LeftLabel and \RightLabel (i.e., the distance to the line is \ScoreOverhang + \labelSpacing.

\def\LeftLineLabel#1{\global\def\displayLeftLineLabel{\llap{#1\hskip\ScoreOverhangLeft\hskip\labelSpacing}}\ignorespaces}

\def\RightLineLabel#1{\global\def\displayRightLineLabel{\rlap{\hskip\ScoreOverhangLeft\hskip\labelSpacing #1}}\ignorespaces}

\global\let\displayLeftLineLabel\relax
\global\let\displayRightLineLabel\relax

3.2 Subproof Labels

Sometimes you'd like to label entire subproofs. This is done with commands \LeftSubproofLabel and \RightSubproofLabel.

\def\LeftSubproofLabel#1{\global\setbox \curBox = \hbox{\vbox to \ht\curBox{\vfil\llap{#1}}}}

7
3.3 Patched commands from bussproofs

Some commands from bussprooﬁngs.sty have to be redeﬁned to include bussprooﬁngs-extra functionality. Added/changed lines are indicated by a %bpextra comment.

\def\resetInferenceDefaults{% 
\global\def\theHypSeparation{\defaultHypSeparation}% 
\global\setbox\myBoxLL=\hbox{\defaultLeftLabel}% 
\global\setbox\myBoxRL=\hbox{\defaultRightLabel}% 
\global\def\buildScore{\alwaysBuildScore}% 
\global\def\theScoreFiller{\alwaysScoreFiller}% 
% reset line labels to nothing %bpextra 
\global\let\displayLeftLineLabel\relax %bpextra 
\global\let\displayRightLineLabel\relax %bpextra 
% reset to default deduce style %bpextra 
\alwaysDeduce %bpextra 
\gdef\hypKernAmt{0pt}% Restore to zero kerning.
}

\def\Axiom$#1\fCenter#2${% 
% Get level and correct names set. 
\prepAxiom% 
% Define the boxes 
% bpextra -- add line labels 
\setbox\myBoxA=\hbox{\displayLeftLineLabel$\mathord{#1}\fCenter\mathord{\relax}$}% %bpextra 
\setbox\myBoxB=\hbox{$#2$\displayRightLineLabel}% %bpextra 
\global\setbox\curBox=\hbox\hspace{\ScoreOverhangLeft}\relax % 
\unhcopy\myBoxA\unhcopy\myBoxB\hspace{\ScoreOverhangRight}\relax% 
% Set the relevant dimensions for the boxes 
\global\curScoreStart=Opt \relax 
\global\curScoreEnd=wd\curBox \relax 
\global\curCenter=wd\myBoxA \relax 
\global\advance\curCenter by \ScoreOverhangLeft% 
\ignorespaces 
}

\def\AxiomC#1{ % Note argument not in math mode 
% Get level and correct names set.
\prepAxiom%
% Define the box.
\setbox\myBoxA=\hbox{\displayLeftLineLabel \#1\displayRightLineLabel}% \bprextra
\global\setbox\curBox =%
\hbox{\hskip\ScoreOverhangLeft\relax%
\unhcopy\myBoxA\hskip\ScoreOverhangRight\relax}%
% Set the relevant dimensions for the boxes
\global\curScoreStart=0pt \relax
\global\curScoreEnd=\wd\curBox \relax
\global\curCenter=.5\wd\curBox \relax
\global\advance \curCenter by \ScoreOverhangLeft%

\def\buildConclusion#1#2{\% Build lower sequent w/ center at $\fCenter$ position.
% Define the boxes
\setbox\myBoxA=\hbox{\displayLeftLineLabel $\mathord{#1}\fCenter\mathord{\relax}$}% \bprextra
\setbox\myBoxB=\hbox{$#2$\displayRightLineLabel}% \bprextra
% Put them together in \myBoxC
\setbox\myBoxC =%
\hbox{\hskip\ScoreOverhangLeft\relax%
\unhcopy\myBoxA\unhcopy\myBoxB\hskip\ScoreOverhangRight\relax}%
% Calculate the center of the \myBoxC string.
\newScoreStart=0pt \relax%
\newCenter=\wd\myBoxA \relax%
\advance \newCenter by \ScoreOverhangLeft%
\newScoreEnd=\wd\myBoxC%
}\%
\def\buildConclusionC#1{\% Build lower sequent w/o $\fCenter$ present.
% Define the box.
\setbox\myBoxA=\hbox{\displayLeftLineLabel #1\displayRightLineLabel}% \bprextra
\setbox\myBoxC =%
\hbox{\hskip\ScoreOverhangLeft\relax%
\unhcopy\myBoxA\hskip\ScoreOverhangRight\relax}%
% Calculate kerning to line up centers
\newScoreStart=0pt \relax%
\newCenter=\wd\myBoxC \relax%
\newScoreEnd=\wd\myBoxC%
\advance \newCenter by \ScoreOverhangLeft%
}\%

\textbf{Change History}

v0.1 \quad \text{dotsdDeduce, added examples} \quad 1
\quad \text{General: Initial version with deduce, linelabel functionality} \quad 1
v0.3 \quad \text{General: Rename to bussproofs-extra.sty} \quad \ldots \ldots \ldots \quad 1
v0.2 \quad \text{General: Fixed bug in}