

The bussproofs-extra package*

Richard Zach

1 Introduction

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

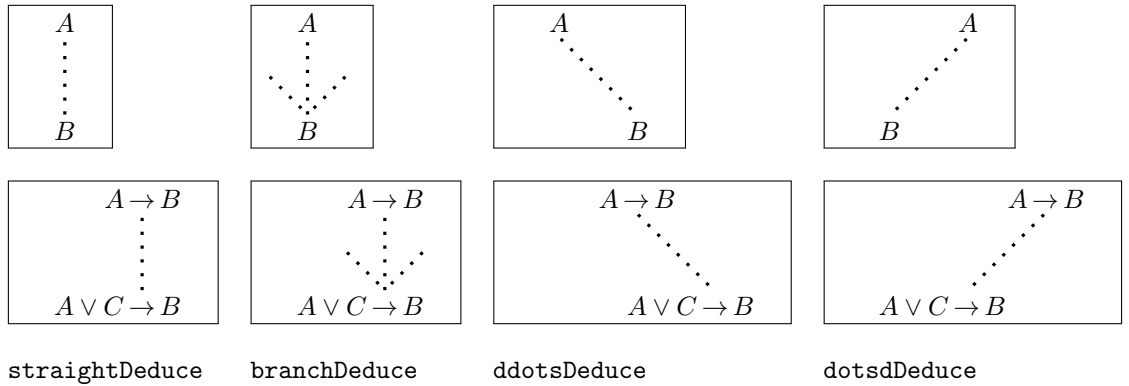
1. `\Deduce$` and `\DeduceC` commands, which work much like `\Infer`, commands but indicate missing parts of a proof.
2. Multiple styles for typesetting the result of `\Deduce`, including
 - (a) `\straightDeduce`, which produces vertical dots
 - (b) `\branchDeduce`, which produces diagonal plus vertical dots
 - (c) `\ddotsDeduce`, which produces diagonal dots from top left to bottom right
 - (d) `\dotsdDeduce`, which produces diagonal dots from top right to bottom left

`\straightDeduce` is the default. It can be changed by redefining `\alwaysDeduce`.

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

Here's what these deductions look like:

*This document corresponds to 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\text{all}\$R\$}
\UnaryInf{\Gamma_1 \fCenter \Theta_1, \forall\text{all } x, F(x)\$}
\ddotsDeduce
\RightLabel{\$\pi_1'\$}
\Deduce{\Gamma \fCenter \Theta, \forall\text{all } x, F(x)\$}
\AxiomC{}
\RightLabel{\$\pi_2\$}
\Deduce{F(n), \Delta_1 \fCenter \Lambda_1\$}
\RightLabel{\$\forall\text{all}\$L\$}
\UnaryInf{\forall\text{all } x, F(x), \Delta_1 \fCenter \Lambda_1\$}
\dotsdDeduce
\RightLabel{\$\pi_2'\$}
\Deduce{\forall\text{all } 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:

$$\begin{array}{c}
\begin{array}{c} \vdots \\ \pi_1(a) \\ \vdots \end{array} \\
\frac{\Gamma_1 \rightarrow \Theta_1, F(a)}{\Gamma_1 \rightarrow \Theta_1, \forall x F(x)} \forall R \\
\begin{array}{c} \vdots \\ \pi_2 \\ \vdots \end{array} \\
\frac{F(n), \Delta_1 \rightarrow \Lambda_1}{\forall x F(x), \Delta_1 \rightarrow \Lambda_1} \forall L \\
\begin{array}{c} \vdots \\ \pi_1' \\ \vdots \end{array} \\
\frac{\Gamma \rightarrow \Theta, \forall x F(x)}{\Gamma, \Delta \rightarrow \Theta, \Lambda} \text{ cut} \\
\begin{array}{c} \vdots \\ \pi_2' \\ \vdots \end{array} \\
\frac{\forall x F(x), \Delta \rightarrow \Lambda}{\Gamma, \Delta \rightarrow \Theta, \Lambda} \text{ cut} \\
\begin{array}{c} \vdots \\ \pi_4 \\ \vdots \end{array} \\
\Pi \rightarrow \Xi
\end{array}$$

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

```

\begin{prooftree}
\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{prooftree}

```

$$\begin{array}{c}
\left(\begin{array}{c} \vdots \\ \Gamma \rightarrow \Delta \\ \vdots \end{array} \right) \pi \\
\left(\begin{array}{c} \vdots \\ \Gamma' \rightarrow \Delta' \\ \vdots \end{array} \right) \pi' \\
\frac{\Gamma \rightarrow \Delta, A \quad A, \Gamma' \rightarrow \Delta'}{\Gamma, \Gamma' \rightarrow \Delta, \Delta'} \text{ cut} \\
\vdots \\
\Pi \rightarrow \Lambda
\end{array}$$

2 Implementation

2.1 Setup

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

```

1 \RequirePackage{bussproofs}
2 \RequirePackage{tikz}
3

```

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.

```

4 \newdimen\CenterCorrection
5 \newdimen\DiagCorrection
6

```

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.

```

7 \def\straightDeduce{%
8   \gdef\fDeduce{\tikz\draw[very thick,loosely dotted] (0,0) -- (0,1);}
9   \global\DiagCorrection=0pt
10  \ignorespaces
11 }
12
13 \def\branchDeduce{%
14   \gdef\fDeduce{\begin{tikzpicture}
15     \draw[very thick,loosely dotted] (0,0) -- (0,1);
16     \draw[very thick,loosely dotted] (-.5,.5) -- (0,0);
17     \draw[very thick,loosely dotted] (.5,.5) -- (0,0);
18   \end{tikzpicture}}
19   \global\DiagCorrection=0pt
20   \ignorespaces
21 }
22
23 \def\ddotsDeduce{%
24   \gdef\fDeduce{\begin{tikzpicture}
25     \draw[very thick,loosely dotted] (0,1) -- (1,0);
26   \end{tikzpicture}}
27   \setbox\myBoxA=\hbox{\fDeduce}
28   \global\DiagCorrection=-\wd\myBoxA
29   \ignorespaces
30 }

```

```

31
32 \def\dotsdDeduce{%
33   \gdef\fDeduce{\begin{tikzpicture}
34     \draw[very thick,loosely dotted] (1,1) -- (0,0);
35   \end{tikzpicture}}
36   \setbox\myBoxA=\hbox{\fDeduce}
37   \global\DiagCorrection=\wd\myBoxA
38   \ignorespaces
39 }
40
41 \def\alwaysDeduce{\straightDeduce}
42 \straightDeduce
43

```

2.4 \backslash Deduce\$ and \backslash DeduceC

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

```

44 \def\Deduce$#1\fCenter#2${%
45   \prepUnary%
46   \buildConclusion{#1}{#2}%
47   \setbox\myBoxA=\hbox{\fCenter}
48   % if we align at \fCenter, move \fDeduce left by 1/2 width of \fCenter
49   \global\CenterCorrection=-.5\wd\myBoxA
50   \joinDeduce%
51   \resetInferenceDefaults%
52   \ignorespaces%
53 }
54
55 \def\DeduceC#1{
56   \prepUnary%
57   \buildConclusionC{#1}%
58   % vdot alignment is off by a bit, correct
59   \global\CenterCorrection=-4pt
60   % Align and join the curBox and the new box into one vbox.
61   \joinDeduce%
62   \resetInferenceDefaults%
63   \ignorespaces%
64 }
65

```

3 Typesetting the Deduction

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

```

66 \def\joinDeduce{%

```

```

67   \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
68   \ifnum\curCenter<\newCenter%
69     \displace=\newCenter%
70     \advance \displace by -\curCenter%
71     \kernUpperBox%
72   \else%
73     \displace=\curCenter%
74     \advance \displace by -\newCenter%
75     \kernLowerBox%
76   \fi%
For \ddotsDeduce, move lower box right; for \dotsoDeduce, move upper box
right; then set \curCenter to align with horizontal center of dots.
77   \ifnum\DiagCorrection<0%
78     \displace=-\DiagCorrection
79     \kernLowerBox%
80   \else
81     \displace=\DiagCorrection
82     \kernUpperBox%
83   \fi%
84   \advance\curCenter by-.5\DiagCorrection
85   %\ifnum \newScoreStart < \curScoreStart %
86   % \global \curScoreStart = \newScoreStart \fi%
87   %\ifnum \curScoreEnd < \newScoreEnd %
88   % \global \curScoreEnd = \newScoreEnd \fi%
89   % Leave room for the left label.
90   %\ifnum \curScoreStart<\wd\myBoxLL%
91   % \global\displace = \wd\myBoxLL%
92   % \global\advance\displace by -\curScoreStart%
93   % \kernUpperBox%
94   % \kernLowerBox%
95   %\fi%
Now we draw the deduction.
96   \buildDeduce%
Put the deduction and labels into a box.
97   \buildScoreLabels%
Put everything into a new box and compute the dimensions for the next \Deduce
or \XxxxInf.
98   \ifx\rootAtBottomFlag\myTrue%
99     \buildRootBottom%
100  \else%
101    \buildRootTop%
102  \fi%
103  \global \curScoreStart=\newScoreStart%
104  \global \curScoreEnd=\newScoreEnd%

```

```

105   \global \curCenter=\newCenter%
106 }
107
\buildDeduce does for \DeduceX what \buildInf does for \XxxInf: put the
deduction bit (dots) into a box and set the dimensions properly.
108 \def\buildDeduce{%
109   \global\setbox \myBoxD =%
110     \hbox{\fDeduce}%
111   \displace = \wd\myBoxD % find width of vdots
set start and end of current score to left and right of the box holding the deduction.
112   \global\curScoreStart = \curCenter%
113   \global\advance\curScoreStart by -.5\displace%
114   \global\curScoreEnd = \curCenter%
115   \global\advance\curScoreEnd by .5\displace%
116   \global\advance\curScoreStart by\CenterCorrection
117   \global\advance\curScoreEnd by\CenterCorrection
118 }
119

```

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.

```

120 \def\LeftLineLabel#1{%
121   \global\def\displayLeftLineLabel{%
122     \llap{#1\hskip\ScoreOverhangLeft\hskip\labelSpacing}}
123   \ignorespaces}
124
125 \def\RightLineLabel#1{%
126   \global\def\displayRightLineLabel{
127     \rlap{\hskip\ScoreOverhangLeft\hskip\labelSpacing #1}}
128   \ignorespaces}
129
130 \global\let\displayLeftLineLabel\relax
131 \global\let\displayRightLineLabel\relax
132

```

3.2 Subproof Labels

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

```

133
134 \def\LeftSubproofLabel#1{%
135   \global\setbox\curBox =
136   \hbox{\vbox to \ht\curBox{\vfil\llap{#1

```

```

137     $\left\{\vrule height .5\ht\curBox width Opt\right.$}\vfil}\box\curBox}%
138 }
139
140 \def\RightSubproofLabel#1{%
141   \displace=\ht\curBox
142   \global\setbox\curBox =
143   \hbox{\box\curBox\vbox to \displace{\vfil
144     \rlap{$\left.\vrule height .5\displace width Opt\right\}$#1}\vfil}}%
145 }
146

```

3.3 Patched commands from bussproofs

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

```

147 \def\resetInferenceDefaults{%
148   \global\def\theHypSeparation{\defaultHypSeparation}%
149   \global\setbox\myBoxLL=\hbox{\defaultLeftLabel}%
150   \global\setbox\myBoxRL=\hbox{\defaultRightLabel}%
151   \global\def\buildScore{\alwaysBuildScore}%
152   \global\def\theScoreFiller{\alwaysScoreFiller}%
153   % reset line labels to nothing %bpextra
154   \global\let\displayLeftLineLabel\relax %bpextra
155   \global\let\displayRightLineLabel\relax %bpextra
156   % reset to default deduce style %bpextra
157   \alwaysDeduce %bpextra
158   \gdef\hypKernAmt{Opt}% Restore to zero kerning.
159 }
160
161 \def\Axiom$#1\fCenter#2${%
162   % Get level and correct names set.
163   \prepAxiom%
164   % Define the boxes
165   % bpextra -- add line labels
166   \setbox\myBoxA=\hbox{\displayLeftLineLabel$\mathord{#1}\fCenter\mathord{\relax}$}% %bpextra
167   \setbox\myBoxB=\hbox{##2$\displayRightLineLabel}% %bpextra
168   \global\setbox\curBox=%
169     \hbox{\hskip\ScoreOverhangLeft\relax%
170       \unhcopy\myBoxA\unhcopy\myBoxB\hskip\ScoreOverhangRight\relax}%
171   % Set the relevant dimensions for the boxes
172   \global\curScoreStart=Opt \relax
173   \global\curScoreEnd=\wd\curBox \relax
174   \global\curCenter=\wd\myBoxA \relax
175   \global\advance \curCenter by \ScoreOverhangLeft%
176   \ignorespaces
177 }
178
179 \def\AxiomC#1{      % Note argument not in math mode
180   % Get level and correct names set.

```



```

181 \prepAxiom%
182 % Define the box.
183 \setbox\myBoxA=\hbox{\displayLeftLineLabel #1\displayRightLineLabel}% %bpextra
184 \global\setbox\curBox =%
185 \hbox{\hskip\ScoreOverhangLeft\relax%
186 \unhcopy\myBoxA\hskip\ScoreOverhangRight\relax}%
187 % Set the relevant dimensions for the boxes
188 \global\curScoreStart=0pt \relax
189 \global\curScoreEnd=\wd\curBox \relax
190 \global\curCenter=.5\wd\curBox \relax
191 \global\advance \curCenter by \ScoreOverhangLeft%
192 \ignorespaces
193 }
194
195 \def\buildConclusion#1#2{% Build lower sequent w/ center at \fCenter position.
196 % Define the boxes
197 \setbox\myBoxA=\hbox{\displayLeftLineLabel $\mathord{#1}\fCenter\mathord{\relax}$}% %bp
198 \setbox\myBoxB=\hbox{#2$\displayRightLineLabel}% %bpextra
199 % Put them together in \myBoxC
200 \setbox\myBoxC =%
201 \hbox{\hskip\ScoreOverhangLeft\relax%
202 \unhcopy\myBoxA\unhcopy\myBoxB\hskip\ScoreOverhangRight\relax}%
203 % Calculate the center of the \myBoxC string.
204 \newScoreStart=0pt \relax%
205 \newCenter=\wd\myBoxA \relax%
206 \advance \newCenter by \ScoreOverhangLeft%
207 \newScoreEnd=\wd\myBoxC%
208 }
209
210 \def\buildConclusionC#1{% Build lower sequent w/o \fCenter present.
211 % Define the box.
212 \setbox\myBoxA=\hbox{\displayLeftLineLabel #1\displayRightLineLabel}% %bpextra
213 \setbox\myBoxC =%
214 \hbox{\hbox{\hskip\ScoreOverhangLeft\relax%
215 \unhcopy\myBoxA\hskip\ScoreOverhangRight\relax}}%
216 % Calculate kerning to line up centers
217 \newScoreStart=0pt \relax%
218 \newCenter=.5\wd\myBoxC \relax%
219 \newScoreEnd=\wd\myBoxC%
220 \advance \newCenter by \ScoreOverhangLeft%
221 }

```

Change History

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