

Logic.sty Package Example

André Platzer

Computer Science Department, Carnegie Mellon University, Pittsburgh, USA
aplatzer@cs.cmu.edu

1 Some Syntax

$$P ::= e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \exists x P \mid \langle \alpha \rangle P$$

2 A Sequent Calculus

$$\begin{array}{ll} (\wedge R) \frac{\Gamma \vdash p, \Delta \quad \Gamma \vdash q, \Delta}{\Gamma \vdash p \wedge q, \Delta} & (\vee R) \frac{\Gamma \vdash p, q, \Delta}{\Gamma \vdash p \vee q, \Delta} \\ (\wedge L) \frac{\Gamma, p, q \vdash \Delta}{\Gamma, p \wedge q \vdash \Delta} & (\vee L) \frac{\Gamma, p \vdash \Delta \quad \Gamma, q \vdash \Delta}{\Gamma, p \vee q \vdash \Delta} \\ (\rightarrow R) \frac{\Gamma, p \vdash q, \Delta}{\Gamma \vdash p \rightarrow q, \Delta} & (\text{cut}) \frac{\Gamma \vdash C, \Delta \quad \Gamma, C \vdash \Delta}{\Gamma \vdash \Delta} \quad (C \text{ is any formula}) \end{array}$$

Rule $\rightarrow R$ has been defined by analogy via `\irlabel{implyr}{ \rightarrow }` and can be used without literally having to define the rule.

3 A Hilbert-style Calculus

$$\begin{array}{ll} (\forall i) (\forall x p(x)) \rightarrow p(e) & (e \text{ is any term}) \\ (\forall \rightarrow) \forall x (p(x) \rightarrow q(x)) \rightarrow (\forall x p(x) \rightarrow \forall x q(x)) \\ (\forall \vee) p \rightarrow \forall x p & (x \notin \text{FV}(p)) \\ (\forall \wedge) \forall x (p(x) \wedge q(x)) \leftrightarrow \forall x p(x) \wedge \forall x q(x) \end{array}$$

Repeating an axiom or proof rule later is easy thanks to `\cinferenceruleQuote`:

$$(\forall i) (\forall x p(x)) \rightarrow p(e) \quad (e \text{ is any term})$$

Likewise `\cinferenceruleQuote` is your friend for repeating a rule:

$$(\wedge R) \frac{\Gamma \vdash p, \Delta \quad \Gamma \vdash q, \Delta}{\Gamma \vdash p \wedge q, \Delta}$$

Also `\dinferencerule` has the same effect as `\cinferencerule` but marks it as a derived axiom or derived proof rule instead of a core rule or axiom. If you

absolutely want to secretly declare a rule first before writing it down (there's a few rare reasons to do so), then `\cinferencerulestore` can store a rule for recall. It's highly recommended to, then, use `\cinferencerulequotedef` at the defining occurrence in order for index magic and hyperreferences to work out best.

Also see `TeX/ruledefs.tex` for more examples with canonical names and ways of including them without the need to declare rules using `\irlabel`. It's best to use the same internal code names in all cases for compatibility.

4 An Example Proof

The rule $\wedge R$ above was nice enough to prove this exciting inference by `\linfer`:

$$\wedge R \frac{x > 0 \vdash x^2 > 0 \quad x > 0 \vdash x \neq 0}{x > 0 \vdash x^2 > 0 \wedge x \neq 0}$$

Longer proofs use `sequentdeduction` which has various formatting options:

Aligned (note ! for branch separator):

$$\begin{array}{c} \wedge L \frac{x > 0, y = 5 \vdash x^2 > 0}{x > 0 \wedge y = 5 \vdash x^2 > 0} \quad \wedge L \frac{x > 0, y = 5 \vdash x \neq 0}{x > 0 \wedge y = 5 \vdash x \neq 0} \\ \wedge R \frac{\quad}{x > 0 \wedge y = 5 \vdash x^2 > 0 \wedge x \neq 0} \\ \rightarrow R \frac{\quad}{\vdash x > 0 \wedge y = 5 \rightarrow x^2 > 0 \wedge x \neq 0} \end{array}$$

Unaligned (note & for branch separator):

$$\begin{array}{c} \wedge L \frac{x > 0, y = 5 \vdash x^2 > 0}{x > 0 \wedge y = 5 \vdash x^2 > 0} \quad \wedge L \frac{x > 0, y = 5 \vdash x \neq 0}{x > 0 \wedge y = 5 \vdash x \neq 0} \\ \wedge R \frac{\quad}{x > 0 \wedge y = 5 \vdash x^2 > 0 \wedge x \neq 0} \\ \rightarrow R \frac{\quad}{\vdash x > 0 \wedge y = 5 \rightarrow x^2 > 0 \wedge x \neq 0} \end{array}$$

Default unaligned (note ! for branch separator):

$$\begin{array}{c} \wedge L \frac{x > 0, y = 5 \vdash x^2 > 0}{x > 0 \wedge y = 5 \vdash x^2 > 0} \quad \wedge L \frac{x > 0, y = 5 \vdash x \neq 0}{x > 0 \wedge y = 5 \vdash x \neq 0} \\ \wedge R \frac{\quad}{x > 0 \wedge y = 5 \vdash x^2 > 0 \wedge x \neq 0} \\ \rightarrow R \frac{\quad}{\vdash x > 0 \wedge y = 5 \rightarrow x^2 > 0 \wedge x \neq 0} \end{array}$$

Closing a proof branch uses `\lclose` which accepts optional arguments:

$$\begin{array}{c} \frac{*}{x > 0 \vdash x^2 > 0} \quad \frac{*}{\text{id } x > 0 \vdash x > 0} \\ \wedge R \frac{\quad}{x > 0 \vdash x^2 > 0 \wedge x > 0} \\ \rightarrow R \frac{\quad}{\vdash x > 0 \rightarrow x^2 > 0 \wedge x > 0} \end{array}$$

The separation between premises of a proof is defined for example like this as in the above example to customize:

`\renewcommand{\linferPremissSeparation}{\hspace{0.8cm}}`

The same command can be used to give a lot of extra space between branches:

$$\frac{\frac{\frac{*}{x > 0 \vdash x^2 > 0}}{\wedge R} \quad \frac{\frac{id}{x > 0 \vdash x > 0}}{\rightarrow R}}{x > 0 \vdash x^2 > 0 \wedge x > 0} \rightarrow R$$

Beamer slides also like the `\begin{sequentdeduction}[array+uncover]` option to animate a sequent calculus proof one step at a time, ideally after an `\uncover<+>{}`.

Some styles such as `array` and `+uncover` obtain improved rendering when formatting proofs right-associative. So the bottom-most proof rule outside as the first rule and then the second proof rule inside its left child:

$$\frac{\frac{id}{A \vdash C, B}}{\neg L} \frac{\neg L}{A, \neg B \vdash C} \frac{\neg L}{A \wedge \neg B \vdash A}$$

Longer proofs benefit from using ① for labels and continuing them later. Longer examples using `logic.sty` can also be found on arXiv.

5 Semantics

The value of term e in state ω for interpretation I is denoted $\omega[e]$. For example, $\omega[e \cdot \tilde{e}] = \omega[e] \cdot \omega[\tilde{e}]$.

The fact that formula P is true in ω is denoted $\omega \models P$. Note that one can configure which notation is used everywhere by passing suitable options to `\usepackage{logic}` or `\usepackage{lsemantics}`. See head of file. For example, $\omega \models P \wedge Q$ iff $\omega \models P$ and $\omega \models Q$.

The set of all states in which formula P is true is denoted $I[P]$. For example, $I[P \wedge Q] = I[P] \cap I[Q]$.

The accessibility relation for program α is denoted $I[\alpha]$, where $(\omega, \nu) \in I[\alpha]$ indicates that final state ν is reachable from initial state ω by running α .