The Logic of Hereditary Harrop Formulas as a Specification Logic for Hybrid

Chelsea Battell and Amy Felty

University of Ottawa

LFMTP-16
Porto, Portugal
June 23, 2016

Hybrid (two-level logical framework)

Object Logic (OL)	▸ judgments defined inductively

Hybrid (two-level logical framework)

Object Logic (OL)	► judgments defined inductively

Reasoning Logic	► Calculus of Inductive Constructions (Coq)

Hybrid (two-level logical framework)

Object Logic (OL)	► judgments defined inductively

Representing Higher-Order Abstract Syntax (HOAS)	► represent OL binders with `lambda` ► define Hybrid terms as de Bruijn terms using `expr`

Reasoning Logic	► Calculus of Inductive Constructions (Coq)

Hybrid (two-level logical framework)

Object Logic (OL)	▶ judgments defined inductively
Specification Logic (SL)	▶ defined as inductive type in Coq ▶ parametric in OL provability
Representing Higher-Order Abstract Syntax (HOAS)	▶ represent OL binders with `lambda` ▶ define Hybrid terms as de Bruijn terms using `expr`
Reasoning Logic	▶ Calculus of Inductive Constructions (Coq)

The Logic of Hereditary Harrop Formulas

$$
\begin{array}{rcl}
G & ::= & \top \mid A \mid G \,\&\, G \mid G \vee G \mid D \longrightarrow G \mid \forall_\tau x.G \mid \exists_\tau x.G \\
D & ::= & A \mid G \longrightarrow D \mid D \,\&\, D \mid \forall_\tau x.D \\
\Gamma & ::= & \emptyset \mid \Gamma, D
\end{array}
$$

[Miller, Nadathur; 2012]

The Logic of Hereditary Harrop Formulas

$$
\begin{array}{rcl}
G & ::= & \top \mid A \mid G \mathbin{\&} G \mid G \vee G \mid D \longrightarrow G \mid \forall_\tau x.G \mid \exists_\tau x.G \\
D & ::= & A \mid G \longrightarrow D \mid D \mathbin{\&} D \mid \forall_\tau x.D \\
\Gamma & ::= & \emptyset \mid \Gamma, D
\end{array}
$$

- τ is restricted to second-order types

[Miller, Nadathur; 2012]

The Logic of Hereditary Harrop Formulas

$$
\begin{array}{rcl}
G & ::= & \top \mid A \mid G \mathbin{\&} G \mid G \vee G \mid D \longrightarrow G \mid \forall_\tau x.G \mid \exists_\tau x.G \\
D & ::= & A \mid G \longrightarrow D \mid D \mathbin{\&} D \mid \forall_\tau x.D \\
\Gamma & ::= & \emptyset \mid \Gamma, D
\end{array}
$$

- τ is restricted to second-order types
- Higher-order in the sense of unrestricted implicational complexity

[Miller, Nadathur; 2012]

Sequents as Dependent Types in Coq

Goal-Reduction Sequent

$\texttt{grseq} : \texttt{context} \to \texttt{oo} \to \texttt{Prop}$

$\Gamma \rhd \beta$ is notation for $\texttt{grseq}\ \Gamma\ \beta$

Backchaining Sequent

$\texttt{bcseq} : \texttt{context} \to \texttt{oo} \to \texttt{atm} \to \texttt{Prop}$

$\Gamma, [\beta] \rhd \alpha$ is notation for $\texttt{bcseq}\ \Gamma\ \beta\ \alpha$

[Wang, Chaudhuri, Gacek, Nadathur; 2012]

Sequents as Dependent Types in Coq

Goal-Reduction Sequent

$\texttt{grseq} : \boxed{\texttt{context}} \rightarrow \texttt{oo} \rightarrow \texttt{Prop}$

$\Gamma \rhd \beta$ is notation for $\texttt{grseq}\ \Gamma\ \beta$

Backchaining Sequent

$\texttt{bcseq} : \boxed{\texttt{context}} \rightarrow \texttt{oo} \rightarrow \texttt{atm} \rightarrow \texttt{Prop}$

$\Gamma, [\beta] \rhd \alpha$ is notation for $\texttt{bcseq}\ \Gamma\ \beta\ \alpha$

[Wang, Chaudhuri, Gacek, Nadathur; 2012]

Sequents as Dependent Types in Coq

Goal-Reduction Sequent

$\texttt{grseq} : \texttt{context} \rightarrow \boxed{\texttt{oo}} \rightarrow \texttt{Prop}$

$\Gamma \rhd \beta$ is notation for $\texttt{grseq}\ \Gamma\ \beta$

Backchaining Sequent

$\texttt{bcseq} : \texttt{context} \rightarrow \boxed{\texttt{oo}} \rightarrow \texttt{atm} \rightarrow \texttt{Prop}$

$\Gamma, [\beta] \rhd \alpha$ is notation for $\texttt{bcseq}\ \Gamma\ \beta\ \alpha$

[Wang, Chaudhuri, Gacek, Nadathur; 2012]

Sequents as Dependent Types in Coq

Goal-Reduction Sequent

$\texttt{grseq} : \texttt{context} \to \texttt{oo} \to \texttt{Prop}$

$\Gamma \rhd \beta$ is notation for $\texttt{grseq}\ \Gamma\ \beta$

Backchaining Sequent

$\texttt{bcseq} : \texttt{context} \to \texttt{oo} \to \boxed{\texttt{atm}} \to \texttt{Prop}$

$\Gamma, [\beta] \rhd \alpha$ is notation for $\texttt{bcseq}\ \Gamma\ \beta\ \alpha$

[Wang, Chaudhuri, Gacek, Nadathur; 2012]

The Specification Logic (SL)

Goal-Reduction Rules

$$\frac{A \mathrel{:-} G \quad \Gamma \rhd G}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_prog} \qquad \frac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn} \qquad \frac{\Gamma \rhd G_1 \quad \Gamma \rhd G_2}{\Gamma \rhd G_1 \mathbin{\&} G_2}\ \text{g\_and}$$

$$\frac{\Gamma\, ,\, D \rhd G}{\Gamma \rhd D \longrightarrow G}\ \text{g\_imp} \qquad \frac{}{\Gamma \rhd \mathtt{T}}\ \text{g\_tt} \qquad \frac{\mathtt{proper}\ E \quad \Gamma \rhd G\ E}{\Gamma \rhd \mathtt{Some}\ G}\ \text{g\_some}$$

$$\frac{\forall (E : \mathtt{expr}\ \ \mathtt{con}), (\mathtt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{All}\ G}\ \text{g\_all} \qquad \frac{\forall (E : \mathtt{X}), (\Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{Allx}\ G}\ \text{g\_allx}$$

Backchaining Rules

$$\frac{}{\Gamma, [\langle\, A \,\rangle] \rhd A}\ \text{b\_match} \qquad \frac{\Gamma, [D_1] \rhd A}{\Gamma, [D_1 \mathbin{\&} D_2] \rhd A}\ \text{b\_and1} \qquad \frac{\Gamma, [D_2] \rhd A}{\Gamma, [D_1 \mathbin{\&} D_2] \rhd A}\ \text{b\_and2}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp} \qquad \frac{\mathtt{proper}\ E \quad \Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{All}\ D] \rhd A}\ \text{b\_all} \qquad \frac{\Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{Allx}\ D] \rhd A}\ \text{b\_allx}$$

$$\frac{\forall (E : \mathtt{expr}\ \ \mathtt{con}), (\mathtt{proper}\ E \to \Gamma, [D\ E] \rhd A)}{\Gamma, [\mathtt{Some}\ D] \rhd A}\ \text{b\_some}$$

The Specification Logic (SL)

Goal-Reduction Rules

$$\frac{A :- G \quad \Gamma \rhd G}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_prog}$$

$$\frac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn}$$

$$\frac{\Gamma \rhd G_1 \quad \Gamma \rhd G_2}{\Gamma \rhd G_1 \,\&\, G_2}\ \text{g\_and}$$

$$\frac{\Gamma\,,\ D \rhd G}{\Gamma \rhd D \longrightarrow G}\ \text{g\_imp}$$

$$\frac{}{\Gamma \rhd \mathtt{T}}\ \text{g\_tt}$$

$$\frac{\mathtt{proper}\ E \quad \Gamma \rhd G\ E}{\Gamma \rhd \mathtt{Some}\ G}\ \text{g\_some}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{All}\ G}\ \text{g\_all}$$

$$\frac{\forall (E : \mathtt{X}), (\Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{Allx}\ G}\ \text{g\_allx}$$

Backchaining Rules

$$\frac{}{\Gamma, [\langle\, A \,\rangle] \rhd A}\ \text{b\_match}$$

$$\frac{\Gamma, [D_1] \rhd A}{\Gamma, [D_1 \,\&\, D_2] \rhd A}\ \text{b\_and1}$$

$$\frac{\Gamma, [D_2] \rhd A}{\Gamma, [D_1 \,\&\, D_2] \rhd A}\ \text{b\_and2}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp}$$

$$\frac{\mathtt{proper}\ E \quad \Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{All}\ D] \rhd A}\ \text{b\_all}$$

$$\frac{\Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{Allx}\ D] \rhd A}\ \text{b\_allx}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma, [D\ E] \rhd A)}{\Gamma, [\mathtt{Some}\ D] \rhd A}\ \text{b\_some}$$

The Specification Logic (SL)

Goal-Reduction Rules

$$\frac{A \mathrel{:-} G \quad \Gamma \rhd G}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_prog} \qquad \boxed{\frac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn}} \qquad \frac{\Gamma \rhd G_1 \quad \Gamma \rhd G_2}{\Gamma \rhd G_1 \mathbin{\&} G_2}\ \text{g\_and}$$

$$\frac{\Gamma\,,\ D \rhd G}{\Gamma \rhd D \longrightarrow G}\ \text{g\_imp} \qquad \frac{}{\Gamma \rhd \mathtt{T}}\ \text{g\_tt} \qquad \frac{\mathtt{proper}\ E \quad \Gamma \rhd G\ E}{\Gamma \rhd \mathtt{Some}\ G}\ \text{g\_some}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{All}\ G}\ \text{g\_all} \qquad \frac{\forall (E : \mathtt{X}), (\Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{Allx}\ G}\ \text{g\_allx}$$

Backchaining Rules

$$\frac{}{\Gamma, [\langle\, A \,\rangle] \rhd A}\ \text{b\_match} \qquad \frac{\Gamma, [D_1] \rhd A}{\Gamma, [D_1 \mathbin{\&} D_2] \rhd A}\ \text{b\_and1} \qquad \frac{\Gamma, [D_2] \rhd A}{\Gamma, [D_1 \mathbin{\&} D_2] \rhd A}\ \text{b\_and2}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp} \qquad \frac{\mathtt{proper}\ E \quad \Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{All}\ D] \rhd A}\ \text{b\_all} \qquad \frac{\Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{Allx}\ D] \rhd A}\ \text{b\_allx}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma, [D\ E] \rhd A)}{\Gamma, [\mathtt{Some}\ D] \rhd A}\ \text{b\_some}$$

The Specification Logic (SL)

Goal-Reduction Rules

$$\frac{A \mathrel{:-} G \quad \Gamma \rhd G}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_prog} \qquad \frac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn} \qquad \frac{\Gamma \rhd G_1 \quad \Gamma \rhd G_2}{\Gamma \rhd G_1 \,\&\, G_2}\ \text{g\_and}$$

$$\frac{\Gamma\,,\ D \rhd G}{\Gamma \rhd D \longrightarrow G}\ \text{g\_imp} \qquad \frac{}{\Gamma \rhd \mathtt{T}}\ \text{g\_tt} \qquad \frac{\mathtt{proper}\ E \quad \Gamma \rhd G\ E}{\Gamma \rhd \mathtt{Some}\ G}\ \text{g\_some}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{All}\ G}\ \text{g\_all} \qquad \frac{\forall (E : \mathtt{X}), (\Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{Allx}\ G}\ \text{g\_allx}$$

Backchaining Rules

$$\frac{}{\Gamma, [\langle\, A \,\rangle] \rhd A}\ \text{b\_match} \qquad \frac{\Gamma, [D_1] \rhd A}{\Gamma, [D_1 \,\&\, D_2] \rhd A}\ \text{b\_and1} \qquad \frac{\Gamma, [D_2] \rhd A}{\Gamma, [D_1 \,\&\, D_2] \rhd A}\ \text{b\_and2}$$

$$\boxed{\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp}} \qquad \frac{\mathtt{proper}\ E \quad \Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{All}\ D] \rhd A}\ \text{b\_all} \qquad \frac{\Gamma, [D\ E] \rhd A}{\Gamma, [\mathtt{Allx}\ D] \rhd A}\ \text{b\_allx}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma, [D\ E] \rhd A)}{\Gamma, [\mathtt{Some}\ D] \rhd A}\ \text{b\_some}$$

Encoding Sequents as Inductive Dependent Types

$$\vdots$$

$$\frac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn}$$

$$\vdots$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp}$$

$$\vdots$$

```
Inductive grseq : context -> oo -> Prop :=
...
| g_dyn :
 forall (L : context) (D : oo) (A : atm),
 elem D L -> bcseq L D A ->
  grseq L (<A>)
...
with bcseq : context -> oo ->  atm -> Prop :=
...
| b_imp :
 forall (L : context) (F G : oo) (A : atm),
 grseq L G -> bcseq L D A ->
  bcseq L (G ---> D) A.
...
```

Sequent Mutual Induction Principle

$$\frac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \textsf{g\_dyn}$$

$$\frac{\forall (E : \texttt{expr con}), (\texttt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \texttt{All}\ G}\ \textsf{g\_all}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \textsf{b\_imp}$$

$$\vdots$$

$$
\begin{aligned}
\texttt{seq\_mutind} : \forall (P_1 &: \texttt{context} \to \texttt{oo} \to \texttt{Prop}) \\
&(P_2 : \texttt{context} \to \texttt{oo} \to \texttt{atm} \to \texttt{Prop}), \\
&(\forall (c : \texttt{context})(o : \texttt{oo})(a : \texttt{atm}), \\
&\quad o \in c \to c, [o] \rhd a \to P_2\ c\ o\ a \to \\
&\quad P_1\ c\ \langle\, a \,\rangle) \to \\
&(\forall (c : \texttt{context})(o : \texttt{expr con} \to \texttt{oo}), \\
&\quad (\forall (e : \texttt{expr con}), \texttt{proper}\ e \to c \rhd o\ e) \to \\
&\quad (\forall (e : \texttt{expr con}), \texttt{proper}\ e \to P_1\ c\ (o\ e) \to \\
&\quad P_1\ c\ (\texttt{All}\ o)) \to \\
&(\forall (c : \texttt{context})(o_1\ o_2 : \texttt{oo})(a : \texttt{atm}), \\
&\quad c \rhd o_1 \to P_1\ c\ o_1 \to \\
&\quad c, [o_2] \rhd a \to P_2\ c\ o_2\ a \to \\
&\quad P_2\ c\ (o_1 \longrightarrow o_2)\ a) \to \\
&\vdots \\
(\forall (c :\ &\texttt{context})(o : \texttt{oo}), \\
&\qquad c \rhd o \to P_1\ c\ o) \land \\
(\forall (c :\ &\texttt{context})(o : \texttt{oo})(a : \texttt{atm}), \\
&\qquad c, [o] \rhd a \to P_2\ c\ o\ a)
\end{aligned}
$$

Sequent Mutual Induction Principle

$$\frac{\boxed{D \in \Gamma} \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn}$$

$$\frac{\forall (E : \texttt{expr con}), (\texttt{proper}\ E \rightarrow \Gamma \rhd G\ E)}{\Gamma \rhd \texttt{All}\ G}\ \text{g\_all}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp}$$

$\vdots$

$$\begin{array}{l}
\texttt{seq_mutind} : \forall (P_1 : \texttt{context} \rightarrow \texttt{oo} \rightarrow \texttt{Prop}) \\
\quad (P_2 : \texttt{context} \rightarrow \texttt{oo} \rightarrow \texttt{atm} \rightarrow \texttt{Prop}), \\
\quad (\forall (c : \texttt{context})(o : \texttt{oo})(a : \texttt{atm}), \\
\quad\quad \boxed{o \in c} \rightarrow c, [o] \rhd a \rightarrow P_2\ c\ o\ a \rightarrow \\
\quad\quad P_1\ c\ \langle\, a \,\rangle) \rightarrow \\
\quad (\forall (c : \texttt{context})(o : \texttt{expr con} \rightarrow \texttt{oo}), \\
\quad\quad (\forall (e : \texttt{expr con}), \texttt{proper}\ e \rightarrow c \rhd o\ e) \rightarrow \\
\quad\quad (\forall (e : \texttt{expr con}), \texttt{proper}\ e \rightarrow P_1\ c\ (o\ e) \rightarrow \\
\quad\quad P_1\ c\ (\texttt{All}\ o)) \rightarrow \\
\quad (\forall (c : \texttt{context})(o_1\ o_2 : \texttt{oo})(a : \texttt{atm}), \\
\quad\quad c \rhd o_1 \rightarrow P_1\ c\ o_1 \rightarrow \\
\quad\quad c, [o_2] \rhd a \rightarrow P_2\ c\ o_2\ a \rightarrow \\
\quad\quad P_2\ c\ (o_1 \longrightarrow o_2)\ a) \rightarrow \\
\quad \vdots \\
(\forall (c : \texttt{context})(o : \texttt{oo}), \\
\quad\quad\quad c \rhd o \rightarrow P_1\ c\ o) \wedge \\
(\forall (c : \texttt{context})(o : \texttt{oo})(a : \texttt{atm}), \\
\quad\quad\quad c, [o] \rhd a \rightarrow P_2\ c\ o\ a)
\end{array}$$

Sequent Mutual Induction Principle

$$\frac{D \in \Gamma \quad \boxed{\Gamma, [D] \rhd A}}{\Gamma \rhd \langle\, A \,\rangle}\ \text{g\_dyn}$$

$$\frac{\forall (E : \mathtt{expr\ \ con}), (\mathtt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{All}\ G}\ \text{g\_all}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \text{b\_imp}$$

$$\vdots$$

$$
\begin{aligned}
\mathtt{seq\_mutind} : \forall (P_1 &: \mathtt{context} \to \mathtt{oo} \to \mathtt{Prop}) \\
&(P_2 : \mathtt{context} \to \mathtt{oo} \to \mathtt{atm} \to \mathtt{Prop}), \\
&(\forall (c : \mathtt{context})(o : \mathtt{oo})(a : \mathtt{atm}), \\
&\quad o \in c \to \boxed{c, [o] \rhd a} \to P_2\ c\ o\ a \to \\
&\quad P_1\ c\ \langle\, a \,\rangle) \to \\
&(\forall (c : \mathtt{context})(o : \mathtt{expr\ \ con} \to \mathtt{oo}), \\
&\quad (\forall (e : \mathtt{expr\ \ con}), \mathtt{proper}\ e \to c \rhd o\ e) \to \\
&\quad (\forall (e : \mathtt{expr\ \ con}), \mathtt{proper}\ e \to P_1\ c\ (o\ e) \to \\
&\quad P_1\ c\ (\mathtt{All}\ o)) \to \\
&(\forall (c : \mathtt{context})(o_1\ o_2 : \mathtt{oo})(a : \mathtt{atm}), \\
&\quad c \rhd o_1 \to P_1\ c\ o_1 \to \\
&\quad c, [o_2] \rhd a \to P_2\ c\ o_2\ a \to \\
&\quad P_2\ c\ (o_1 \longrightarrow o_2)\ a) \to \\
&\quad \vdots \\
(\forall (c :\ &\mathtt{context})(o : \mathtt{oo}), \\
&\qquad c \rhd o \to P_1\ c\ o) \land \\
(\forall (c :\ &\mathtt{context})(o : \mathtt{oo})(a : \mathtt{atm}), \\
&\qquad c, [o] \rhd a \to P_2\ c\ o\ a)
\end{aligned}
$$

Sequent Mutual Induction Principle

$$\frac{D \in \Gamma \quad \boxed{\Gamma, [D] \rhd A}}{\Gamma \rhd \langle\, A \,\rangle}\ \texttt{g\_dyn}$$

$$\frac{\forall (E : \texttt{expr}\ \ \texttt{con}), (\texttt{proper}\ E \to \Gamma \rhd G\ E)}{\Gamma \rhd \texttt{All}\ G}\ \texttt{g\_all}$$

$$\frac{\Gamma \rhd G \quad \Gamma, [D] \rhd A}{\Gamma, [G \longrightarrow D] \rhd A}\ \texttt{b\_imp}$$

$$\vdots$$

$$\begin{aligned}
\texttt{seq_mutind} : \forall &(P_1 : \texttt{context} \to \texttt{oo} \to \texttt{Prop}) \\
&(P_2 : \texttt{context} \to \texttt{oo} \to \texttt{atm} \to \texttt{Prop}), \\
&(\forall (c : \texttt{context})(o : \texttt{oo})(a : \texttt{atm}), \\
&\quad o \in c \to \boxed{c, [o] \rhd a} \to \boxed{P_2\ c\ o\ a} \to \\
&\quad P_1\ c\ \langle\, a \,\rangle) \to \\
&(\forall (c : \texttt{context})(o : \texttt{expr}\ \ \texttt{con} \to \texttt{oo}), \\
&\quad (\forall (e : \texttt{expr}\ \ \texttt{con}), \texttt{proper}\ e \to c \rhd o\ e) \to \\
&\quad (\forall (e : \texttt{expr}\ \ \texttt{con}), \texttt{proper}\ e \to P_1\ c\ (o\ e) \to \\
&\quad P_1\ c\ (\texttt{All}\ o)) \to \\
&(\forall (c : \texttt{context})(o_1\ o_2 : \texttt{oo})(a : \texttt{atm}), \\
&\quad c \rhd o_1 \to P_1\ c\ o_1 \to \\
&\quad c, [o_2] \rhd a \to P_2\ c\ o_2\ a \to \\
&\quad P_2\ c\ (o_1 \longrightarrow o_2)\ a) \to \\
&\quad \vdots \\
&(\forall (c : \texttt{context})(o : \texttt{oo}), \\
&\qquad\qquad c \rhd o \to P_1\ c\ o) \land \\
&(\forall (c : \texttt{context})(o : \texttt{oo})(a : \texttt{atm}), \\
&\qquad\qquad c, [o] \rhd a \to P_2\ c\ o\ a)
\end{aligned}$$

Generalized Specification Logic (GSL)

All rules of the SL have one of the following forms:

$$\dfrac{\begin{array}{c}\overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \forall\overline{(x_{n,s_n} : R_{n,s_n})},(c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \forall\overline{(y_{p,t_p} : S_{p,t_p})},(c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p})\end{array}}{c \rhd o}\ \text{gr\_rule}$$

$$\dfrac{\begin{array}{c}\overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \forall\overline{(x_{n,s_n} : R_{n,s_n})},(c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \forall\overline{(y_{p,t_p} : S_{p,t_p})},(c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p})\end{array}}{c, [o] \rhd a}\ \text{bc\_rule}$$

SL Rules from GSL Rules

Rule	m	n	p	c	o
$\dfrac{\forall(E : \mathtt{X}), (\Gamma \rhd G\ E)}{\Gamma \rhd \mathtt{Allx}\ G}$ g_allx	0	1	0	Γ	$\mathtt{Allx}\ G$

$$s_1 := 1$$
$$x_{1,1} := E$$
$$R_{1,1} := \mathtt{X}$$
$$\gamma_1 \langle\!\langle \mathtt{Allx}\ G \rangle\!\rangle := \emptyset$$
$$F_1 \langle\!\langle \mathtt{Allx}\ G, E \rangle\!\rangle := G\ E$$

Structural Rules

$$\frac{\Gamma \rhd \beta_2}{\Gamma\, , \beta_1 \rhd \beta_2}\ \texttt{gr\_weakening} \qquad \frac{\Gamma, [\beta_2] \rhd \alpha}{\Gamma\, , \beta_1, [\beta_2] \rhd \alpha}\ \texttt{bc\_weakening}$$

$$\frac{\Gamma\, ,\ \beta_1\, , \beta_1 \rhd \beta_2}{\Gamma\, , \beta_1 \rhd \beta_2}\ \texttt{gr\_contraction} \qquad \frac{\Gamma\, ,\ \beta_1\, ,\ \beta_1, [\beta_2] \rhd \alpha}{\Gamma\, ,\ \beta_1, [\beta_2] \rhd \alpha}\ \texttt{bc\_contraction}$$

$$\frac{\Gamma\, ,\ \beta_2\, ,\ \beta_1 \rhd \beta_3}{\Gamma\, , \beta_1\, ,\ \beta_2 \rhd \beta_3}\ \texttt{gr\_exchange} \qquad \frac{\Gamma\, ,\ \beta_2\, ,\ \beta_1, [\beta_3] \rhd \alpha}{\Gamma\, ,\ \beta_1\, ,\ \beta_2, [\beta_3] \rhd \alpha}\ \texttt{bc\_exchange}$$

These are all corollaries of a general theorem:

Theorem (`monotone`)

$$\frac{\Gamma \subseteq \Gamma' \quad \Gamma \rhd \beta}{\Gamma' \rhd \beta} \land \frac{\Gamma \subseteq \Gamma' \quad \Gamma, [\beta] \rhd \alpha}{\Gamma', [\beta] \rhd \alpha}$$

Theorem (monotone)

$$
\begin{aligned}
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo}),\\
&\qquad \Gamma \rhd \beta \to \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma' \rhd \beta\ \ ) \land\\
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm}),\\
&\qquad \Gamma, [\beta] \rhd \alpha \to \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma', [\beta] \rhd \alpha\ \ )
\end{aligned}
$$

Define

$$
\begin{aligned}
P_1 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})\ .\\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma' \rhd \beta\\
P_2 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm})\ .\\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma', [\beta] \rhd \alpha
\end{aligned}
$$

Proof

By induction over $\Gamma \rhd \beta$ and $\Gamma, [\beta] \rhd \alpha$ using `seq_mutind`.

Theorem (`monotone`)

$$\begin{aligned}&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo}),\\ &\qquad \Gamma \rhd \beta \to \boxed{\forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma' \rhd \beta}\ ) \land\\ &(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm}),\\ &\qquad \Gamma, [\beta] \rhd \alpha \to \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma', [\beta] \rhd \alpha\ \ )\end{aligned}$$

Define

$$\begin{aligned}P_1 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})\ .\\ &\qquad \boxed{\forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma' \rhd \beta}\\ P_2 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm})\ .\\ &\qquad \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma', [\beta] \rhd \alpha\end{aligned}$$

Proof

By induction over $\Gamma \rhd \beta$ and $\Gamma, [\beta] \rhd \alpha$ using `seq_mutind`.

Theorem (monotone)

$$
\begin{aligned}
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo}),\\
&\qquad \Gamma \rhd \beta \to \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma' \rhd \beta\ \ ) \land\\
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm}),\\
&\qquad \Gamma, [\beta] \rhd \alpha \to \boxed{\forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma', [\beta] \rhd \alpha}\ )
\end{aligned}
$$

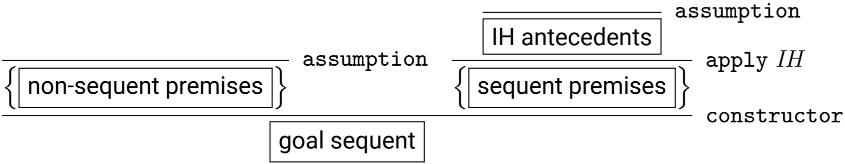
Define

$$
\begin{aligned}
P_1 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})\ .\\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma' \rhd \beta\\
P_2 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm})\ .\\
&\qquad \boxed{\forall(\Gamma' : \texttt{context}), \Gamma \subseteq \Gamma' \to \Gamma', [\beta] \rhd \alpha}
\end{aligned}
$$

Proof

By induction over $\Gamma \rhd \beta$ and $\Gamma, [\beta] \rhd \alpha$ using `seq_mutind`.

Proof Outline for `monotone`



Proof with 15 subcases proven automatically in Coq

```
Proof.
Hint Resolve context_sub_sup.
eapply seq_mutind; intros;
try (econstructor; eauto; eassumption).
Qed.
```

Cut Admissibility

Theorem (`cut_admissible`)

$$\frac{\Gamma, \delta \rhd \beta \qquad \Gamma \rhd \delta}{\Gamma \rhd \beta} \wedge \frac{\Gamma, \delta, [\beta] \rhd \alpha \qquad \Gamma \rhd \delta}{\Gamma, [\beta] \rhd \alpha}$$

Proof by nested induction over δ then mutual structural induction over $\Gamma, \delta \rhd \beta$ and $\Gamma, \delta, [\beta] \rhd \alpha$

[Pfenning; 2000]

Theorem (`cut_admissible`)

$$
\begin{aligned}
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo}), \Gamma \rhd \beta \to \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma' \rhd \beta\ \ ) \wedge \\
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm}), \Gamma, [\beta] \rhd \alpha \to \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma', [\beta] \rhd \alpha\ \ )
\end{aligned}
$$

Define

$$
\begin{aligned}
P_1 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})\ . \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma' \rhd \beta \\
P_2 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm})\ . \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma', [\beta] \rhd \alpha
\end{aligned}
$$

Theorem (`cut_admissible`)

$$
\begin{aligned}
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo}), \Gamma \rhd \beta \to \\
&\qquad \boxed{\forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma' \rhd \beta}\ ) \land \\
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm}), \Gamma, [\beta] \rhd \alpha \to \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma', [\beta] \rhd \alpha\ \ )
\end{aligned}
$$

Define

$$
\begin{aligned}
P_1 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})\ . \\
&\qquad \boxed{\forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma' \rhd \beta} \\
P_2 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm})\ . \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma', [\beta] \rhd \alpha
\end{aligned}
$$

Theorem (`cut_admissible`)

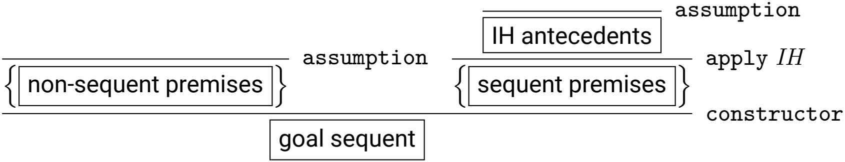
$$\begin{aligned}
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo}), \Gamma \rhd \beta \to \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma' \rhd \beta\ )\ \wedge \\
&(\ \forall(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm}), \Gamma, [\beta] \rhd \alpha \to \\
&\qquad \boxed{\forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma', [\beta] \rhd \alpha}\ )
\end{aligned}$$

Define

$$\begin{aligned}
P_1 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})\ . \\
&\qquad \forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma' \rhd \beta \\
P_2 &:= \lambda(\Gamma : \texttt{context})(\beta : \texttt{oo})(\alpha : \texttt{atm})\ . \\
&\qquad \boxed{\forall(\Gamma' : \texttt{context}), \Gamma = (\Gamma', \delta) \to \Gamma' \rhd \delta \to \Gamma', [\beta] \rhd \alpha}
\end{aligned}$$

Proof Outline for `cut_admissible`

98 of 105 cases proven automatically in Coq




```
Proof.
Hint Resolve gr_weakening context_swap.
induction delta; eapply seq_mutind; intros;
subst; try (econstructor; eauto; eassumption).
...
```

Structural Induction over GSL

Suppose we wish to prove

$$
\begin{aligned}
&(\forall\ (c : \texttt{context})\ (o : \texttt{oo}),\\
&\qquad (\ c \rhd o\ ) \rightarrow (P_1\ c\ o))\ \ \wedge\\
&(\forall\ (c : \texttt{context})\ (o : \texttt{oo})\ (a : \texttt{atm}),\\
&\qquad (\ c, [o] \rhd a\ ) \rightarrow (P_2\ c\ o\ a))
\end{aligned}
$$

for some

$$
\begin{aligned}
&P_1 := \lambda(c : \texttt{context})\ (o : \texttt{oo})\ .\\
&\quad \forall(\Gamma' : \texttt{context}), IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle \rightarrow \cdots \rightarrow IA_w\langle\!\langle c, \Gamma'\rangle\!\rangle \rightarrow \underline{\Gamma' \rhd o}\\
&P_2 := \lambda(c : \texttt{context})\ (o : \texttt{oo})\ (a : \texttt{atm})\ .\\
&\quad \forall(\Gamma' : \texttt{context}), IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle \rightarrow \cdots \rightarrow IA_w\langle\!\langle c, \Gamma'\rangle\!\rangle \rightarrow \underline{\Gamma', [o] \rhd a}
\end{aligned}
$$

by induction over $c \rhd o$ and $c, [o] \rhd a$

Structural Induction over GSL

$$\frac{\begin{array}{r l}\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \triangleright \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \triangleright \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p}\end{array}}{P_1\ c\ o}$$

Structural Induction over GSL

$$\frac{\begin{array}{rl}\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \triangleright \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \triangleright \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p}\end{array}}{P_1\ c\ o}$$

Next: unfold P_1 in goal

Structural Induction over GSL

$$\frac{\begin{array}{rl} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \end{array}}{\begin{array}{l} \forall(\Gamma' : \texttt{context}), IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle \to \cdots \to \\ \qquad IA_w\langle\!\langle c, \Gamma'\rangle\!\rangle \to \Gamma' \rhd o \end{array}}$$

Structural Induction over GSL

$$\frac{\begin{array}{r@{\,}l} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \end{array}}{\begin{array}{c} \forall(\Gamma' : \texttt{context}), IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle \to \cdots \to \\ IA_w\langle\!\langle c, \Gamma'\rangle\!\rangle \to \Gamma' \rhd o \end{array}}$$

Next: introduce induction assumptions

Structural Induction over GSL

$$\frac{\begin{array}{r l} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\ \overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \triangleright \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \triangleright \overline{a_p}) \\ \overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma' \rangle\!\rangle \end{array}}{\Gamma' \triangleright o}$$

Structural Induction over GSL

$$\frac{\begin{array}{rl}\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma'\rangle\!\rangle \end{array}}{\Gamma' \rhd o}$$

Next: backchain with gr_rule

Structural Induction over GSL

$$\frac{\begin{array}{rl} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\ \overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma' \rangle\!\rangle \end{array}}{\begin{array}{l} \overline{Q_m}\langle\!\langle \Gamma', o \rangle\!\rangle, \\ \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (\Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle), \\ \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (\Gamma' \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \end{array}}$$

Structural Induction over GSL

$$
\frac{\begin{array}{r l}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\
\overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1 \ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle) \ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2 \ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle) \ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle) \ \overline{a_p} \\
\overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma' \rangle\!\rangle
\end{array}}{\begin{array}{l}
\overline{Q_m}\langle\!\langle \Gamma', o \rangle\!\rangle, \\
\forall \overline{(x_{n,s_n} : R_{n,s_n})}, \boxed{(\Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle)}, \\
\forall \overline{(y_{p,t_p} : S_{p,t_p})}, (\Gamma' \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p})
\end{array}}
$$

Structural Induction over GSL

$$
\frac{
\begin{array}{r l}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\
\overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\
\overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma' \rangle\!\rangle
\end{array}
}{
\begin{array}{l}
\overline{Q_m}\langle\!\langle \Gamma', o \rangle\!\rangle, \\
\forall \overline{(x_{n,s_n} : R_{n,s_n})}, \boxed{(\Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle)}, \\
\forall \overline{(y_{p,t_p} : S_{p,t_p})}, (\Gamma' \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p})
\end{array}
}
$$

Next: apply induction hypothesis to sequent subgoals

Structural Induction over GSL: Sequent Premises

$$
\frac{
\begin{array}{r l}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\
\overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \triangleright \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \triangleright \overline{a_p}) \\
\overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\
\overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma' \rangle\!\rangle
\end{array}
}{
\overline{IA_w}\langle\!\langle c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle, \Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rangle\!\rangle
}
$$

Structural Induction over GSL: Sequent Premises

$$\dfrac{\begin{array}{rl} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\ \overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma' \rangle\!\rangle \end{array}}{\overline{IA_w}\langle\!\langle c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle, \Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rangle\!\rangle}$$

Proof for `monotone`:

$IA_1\langle\!\langle c, \Gamma' \rangle\!\rangle := c \subseteq \Gamma'$

Structural Induction over GSL: Sequent Premises

$$\cfrac{\begin{array}{r l} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\ IP_1 & : c \subseteq \Gamma' \end{array}}{c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \subseteq \Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle}$$

Proof for `monotone`:

$$IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle \coloneqq c \subseteq \Gamma'$$

Structural Induction over GSL: Sequent Premises

$$
\begin{array}{r l}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\
\overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\
IP_1 & : c \subseteq \Gamma' \\
\hline
& c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \subseteq \Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle
\end{array}
$$

Proof for `monotone`:

Backchain with `context_sub_sup`

Structural Induction over GSL: Sequent Premises

$$
\frac{\begin{array}{rl}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\
\overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\
\overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\
\overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\
IP_1 & : c \subseteq \Gamma'
\end{array}}{c \subseteq \Gamma'}
$$

Proof for `monotone`:

matches assumption IP_1

Structural Induction over GSL: Sequent Premises

$$\cfrac{\begin{array}{rl}\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \triangleright \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \triangleright \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle c, \Gamma'\rangle\!\rangle \end{array}}{\overline{IA_w}\langle\!\langle c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle, \Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle\rangle\!\rangle}$$

Proof for `cut_admissibility`

$IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle := (c = \Gamma', \delta)$ and $IA_2\langle\!\langle c, \Gamma'\rangle\!\rangle := \Gamma' \triangleright \delta$

Structural Induction over GSL: Sequent Premises

$$
\dfrac{
\begin{array}{r l}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\
\overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\
\overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\
\overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\
IP_1 & : c = \Gamma', \delta \\
IP_2 & : \Gamma' \rhd \delta
\end{array}
}{
(c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle = \Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle, \delta), (\Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \delta)
}
$$

Proof for `cut_admissibility`

$IA_1\langle\!\langle c, \Gamma'\rangle\!\rangle := (c = \Gamma', \delta)$ and $IA_2\langle\!\langle c, \Gamma'\rangle\!\rangle := \Gamma' \rhd \delta$

Structural Induction over GSL: Sequent Premises

$$\frac{\begin{array}{rl}\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\ IP_1 & : c = \Gamma', \delta \\ IP_2 & : \Gamma' \rhd \delta \end{array}}{(c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle = \Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle, \delta), (\Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \delta)}$$

Proof for `cut_admissibility`

Sequent subgoal: backchain with `weakening`

Structural Induction over GSL: Sequent Premises

$$\dfrac{\begin{array}{r l} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\ \overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\ \overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\ IP_1 & : c = \Gamma', \delta \\ IP_2 & : \Gamma' \rhd \delta \end{array}}{(c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle = \Gamma' \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle, \delta), (\Gamma' \rhd \delta)}$$

Proof for `cut_admissibility`

Sequent subgoal: matches assumption IP_2

Structural Induction over GSL: Sequent Premises

$$
\frac{
\begin{array}{rl}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\
\overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\
\overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\
IP_1 & : c = \Gamma', \delta \\
IP_2 & : \Gamma' \rhd \delta
\end{array}
}{
(c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle = \Gamma' \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle, \delta)
}
$$

Proof for `cut_admissibility`

Context equality subgoal: backchain with `context_swap`

Structural Induction over GSL: Sequent Premises

$$
\begin{array}{rl}
\overline{H_m} & : \overline{Q_m}\langle\!\langle c, o\rangle\!\rangle \\
\overline{Hg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\
\overline{IHg_n} & : \forall\overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}}\rangle\!\rangle) \\
\overline{Hb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle] \rhd \overline{a_p}) \\
\overline{IHb_p} & : \forall\overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o\rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}}\rangle\!\rangle)\ \overline{a_p} \\
IP_1 & : c = \Gamma', \delta \\
IP_2 & : \Gamma' \rhd \delta \\
\hline
\multicolumn{2}{c}{(c \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle = (\Gamma', \delta) \cup \overline{\gamma_n}\langle\!\langle o\rangle\!\rangle)}
\end{array}
$$

Proof for `cut_admissibility`

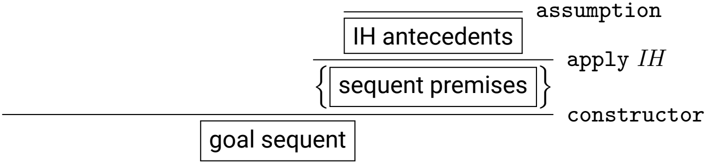
Context equality subgoal: backchain with `context_sub_sup`

Structural Induction over GSL: Sequent Premises

$$\frac{\begin{array}{rl} \overline{H_m} & : \overline{Q_m}\langle\!\langle c, o \rangle\!\rangle \\ \overline{Hg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle \rhd \overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{IHg_n} & : \forall \overline{(x_{n,s_n} : R_{n,s_n})}, P_1\ (c \cup \overline{\gamma_n}\langle\!\langle o \rangle\!\rangle)\ (\overline{F_n}\langle\!\langle o, \overline{x_{n,s_n}} \rangle\!\rangle) \\ \overline{Hb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle, [\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle] \rhd \overline{a_p}) \\ \overline{IHb_p} & : \forall \overline{(y_{p,t_p} : S_{p,t_p})}, P_2\ (c \cup \overline{\gamma'_p}\langle\!\langle o \rangle\!\rangle)\ (\overline{F'_p}\langle\!\langle o, \overline{y_{p,t_p}} \rangle\!\rangle)\ \overline{a_p} \\ IP_1 & : c = \Gamma', \delta \\ IP_2 & : \Gamma' \rhd \delta \end{array}}{c = \Gamma', \delta}$$

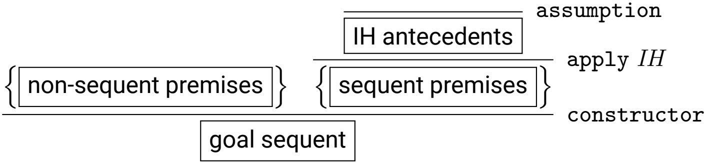
Proof for `cut_admissibility`

matches assumption IP_1



Completed:

- branches of proof for sequent premises



Completed:

- branches of proof for sequent premises

To Do:

- branches for non-sequent premises

Structural Induction over GSL: Non-sequent Premises

Case $\dfrac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\ A\ \rangle}$ **g_dyn:**

$$\begin{array}{rl} H_1 & : D \in \Gamma \\ Hb_1 & : \Gamma, [D] \rhd a_1 \\ IHb_1 & : P_2\ \Gamma\ D\ a_1 \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle \Gamma, \Gamma' \rangle\!\rangle \\ \hline & D \in \Gamma' \end{array}$$

Structural Induction over GSL: Non-sequent Premises

Case $\dfrac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\ A\ \rangle}$ **g_dyn:**

$$\dfrac{\begin{array}{rl} H_1 & : D \in \Gamma \\ Hb_1 & : \Gamma, [D] \rhd a_1 \\ IHb_1 & : P_2\ \Gamma\ D\ a_1 \\ \overline{IP_w} & : \overline{IA_w}\langle\!\langle \Gamma, \Gamma' \rangle\!\rangle \end{array}}{D \in \Gamma'}$$

Proof for `monotone`**:**

$IA_1\langle\!\langle \Gamma, \Gamma' \rangle\!\rangle \coloneqq \Gamma \subseteq \Gamma'$

Structural Induction over GSL: Non-sequent Premises

Case $\dfrac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\ A\ \rangle}$ g_dyn**:**

$$\dfrac{\begin{array}{r l} H_1 & : D \in \Gamma \\ Hb_1 & : \Gamma, [D] \rhd a_1 \\ IHb_1 & : P_2\ \Gamma\ D\ a_1 \\ IP_1 & : \Gamma \subseteq \Gamma' \end{array}}{D \in \Gamma'}$$

Proof for `monotone`:

$IA_1 \langle\!\langle \Gamma, \Gamma' \rangle\!\rangle \coloneqq \Gamma \subseteq \Gamma'$

Structural Induction over GSL: Non-sequent Premises

Case $\dfrac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\ A\ \rangle}$ **g_dyn:**

$$\dfrac{\begin{array}{r l} H_1 & : D \in \Gamma \\ Hb_1 & : \Gamma, [D] \rhd a_1 \\ IHb_1 & : P_2\ \Gamma\ D\ a_1 \\ IP_1 & : \Gamma \subseteq \Gamma' \end{array}}{D \in \Gamma'}$$

Proof for `monotone`**:**

Unfold $\subseteq$ in IP_1

Structural Induction over GSL: Non-sequent Premises

Case $\dfrac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\ A\ \rangle}$ **g_dyn:**

$$\dfrac{\begin{aligned} H_1 &: D \in \Gamma \\ Hb_1 &: \Gamma, [D] \rhd a_1 \\ IHb_1 &: P_2\ \Gamma\ D\ a_1 \\ IP_1 &: \forall (o : \mathsf{oo}), o \in \Gamma \to o \in \Gamma' \end{aligned}}{D \in \Gamma'}$$

Proof for `monotone`**:**

Backchain over IP_1

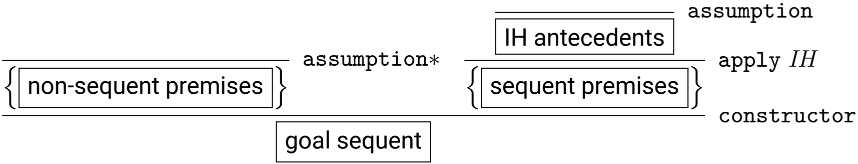
Structural Induction over GSL: Non-sequent Premises

Case $\dfrac{D \in \Gamma \quad \Gamma, [D] \rhd A}{\Gamma \rhd \langle\ A\ \rangle}$ **g_dyn:**

$$\dfrac{\begin{array}{rl} H_1 & : D \in \Gamma \\ Hb_1 & : \Gamma, [D] \rhd a_1 \\ IHb_1 & : P_2\ \Gamma\ D\ a_1 \\ IP_1 & : \forall (o : \mathsf{oo}), o \in \Gamma \to o \in \Gamma' \end{array}}{D \in \Gamma}$$

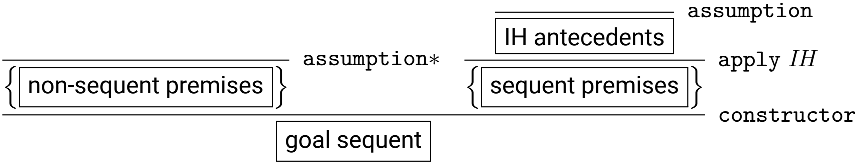
Proof for `monotone`:

Matches H_1



Completed:

- branches of proof for sequent premises
- branches of proof for non-sequent premises for `monotone`



Completed:

- branches of proof for sequent premises
- branches of proof for non-sequent premises for `monotone`

Not shown:

- `cut_admissibility` subcases for rules with non-sequent premises (see paper)

Future Work

- Case studies to illustrate the benefit of the new SL

Future Work

- Case studies to illustrate the benefit of the new SL
  1. correspondence between HOAS and de Bruijn encodings of untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]

Future Work

- Case studies to illustrate the benefit of the new SL
  1. correspondence between HOAS and de Bruijn encodings of untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
  2. structural characterization of reductions on untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]

Future Work

- Case studies to illustrate the benefit of the new SL
  1. correspondence between HOAS and de Bruijn encodings of untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
  2. structural characterization of reductions on untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
  3. algorithmic specification of bounded subtype polymorphism in System F [Pientka; 2007]

Future Work

- Case studies to illustrate the benefit of the new SL
  1. correspondence between HOAS and de Bruijn encodings of untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
  2. structural characterization of reductions on untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
  3. algorithmic specification of bounded subtype polymorphism in System F [Pientka; 2007]
- Apply generalized SL to other logics and proof to other metatheorems

Future Work

- Case studies to illustrate the benefit of the new SL
    1. correspondence between HOAS and de Bruijn encodings of untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
    2. structural characterization of reductions on untyped λ-terms [Wang, Chaudhuri, Gacek, Nadathur; 2012]
    3. algorithmic specification of bounded subtype polymorphism in System F [Pientka; 2007]
- Apply generalized SL to other logics and proof to other metatheorems
- Compare cut admissibility proof here to Abella

Conclusions

- Add SL based on hereditary Harrop formulas to Hybrid

Conclusions

- Add SL based on hereditary Harrop formulas to Hybrid
- Prove structural properties of new SL

Conclusions

- Add SL based on hereditary Harrop formulas to Hybrid
- Prove structural properties of new SL
- Generalization of SL rules

Conclusions

- Add SL based on hereditary Harrop formulas to Hybrid
- Prove structural properties of new SL
- Generalization of SL rules
- Proof by structural induction over generalized SL (encapsulate collections of cases into one)

Conclusions

- Add SL based on hereditary Harrop formulas to Hybrid
- Prove structural properties of new SL
- Generalization of SL rules
- Proof by structural induction over generalized SL (encapsulate collections of cases into one)

Thank you!