

Adjunction Based Categorical Logic Programming

Ed Morehouse

Wesleyan University

March 30, 2012

Outline

- 1 A Brief History of Logic Programming
- 2 Proof Search in Proof Theory
- 3 Adjunctions in Categorical Proof Theory
- 4 Connective Chirality and Search Strategy

1 A Brief History of Logic Programming

In the beginning there was PROLOG

PROLOG has:

- the logic of Horn clauses
- operational semantics of resolution
- denotational semantics based in model theory (classical logic)

Weaknesses:

- to be practical, incorporates extra-logical features (e.g. assert/retract, cut)
- not clear how to extend to other logics

“logic programming is logic plus control” – Kowalski

How to get the logic to do more of the work?

In the beginning there was PROLOG

PROLOG has:

- the logic of Horn clauses
- operational semantics of resolution
- denotational semantics based in model theory (classical logic)

Weaknesses:

- to be practical, incorporates extra-logical features (e.g. assert/retract, cut)
- not clear how to extend to other logics

“logic programming is logic plus control” – Kowalski

How to get the logic to do more of the work?

In the beginning there was PROLOG

PROLOG has:

- the logic of Horn clauses
- operational semantics of resolution
- denotational semantics based in model theory (classical logic)

Weaknesses:

- to be practical, incorporates extra-logical features (e.g. assert/retract, cut)
- not clear how to extend to other logics

“logic programming is logic plus control” – Kowalski

How to get the logic to do more of the work?

The Miller Revolution

Dale Miller realized the connective of implication could interpret scoping and modules [Mil89]:

The goal $D \supset G$ could mean, “try to *satisfy* the goal G in the current environment augmented by the local assumption D ”.

But not classically:

$$G1 \vee (D \supset G2) \equiv_c G1 \vee \neg D \vee G2 \equiv_c (D \supset G1) \vee G2 \equiv_c (D \supset G1) \vee (D \supset G2)$$

Intuitionistic logic makes a natural choice.

The Miller Revolution

Dale Miller realized the connective of implication could interpret scoping and modules [Mil89]:

The goal $D \supset G$ could mean, “try to *satisfy* the goal G in the current environment augmented by the local assumption D ”.

But not classically:

$$G1 \vee (D \supset G2) \equiv_c G1 \vee \neg D \vee G2 \equiv_c (D \supset G1) \vee G2 \equiv_c (D \supset G1) \vee (D \supset G2)$$

Intuitionistic logic makes a natural choice.

Abstract Logic Programming Languages

Miller et al [Mil+91] proposed an operational semantics for **goal-directed proof search** in *intuitionistic logic* by taking *inversions* of natural deduction introduction rules (sequent right rules) as **search instructions** (\vdash_o):

SUCCESS : $\Gamma \vdash_o \top$ always

FAILURE : $\Gamma \vdash_o \perp$ never

BOTH : $\Gamma \vdash_o A \wedge B$ only if $\Gamma \vdash_o A$ and $\Gamma \vdash_o B$

EITHER : $\Gamma \vdash_o A \vee B$ only if $\Gamma \vdash_o A$ or $\Gamma \vdash_o B$

AUGMENT : $\Gamma \vdash_o A \supset B$ only if $\Gamma, A \vdash_o B$

GENERIC : $\Gamma \vdash_o \forall x : X. A$ only if $\Gamma \vdash_o A [x \mapsto e]$ for $e \notin FV(\Gamma, A)$

INSTANCE : $\Gamma \vdash_o \exists x : X. A$ only if $\Gamma \vdash_o A [x \mapsto t]$ for some $t : X$

Abstract Logic Programming Languages

Miller et al [Mil+91] proposed an operational semantics for **goal-directed proof search** in *intuitionistic logic* by taking *inversions* of natural deduction introduction rules (sequent right rules) as **search instructions** (\vdash_o):

- Easy to understand and implement
- Not complete (eg: $A \vee B \not\vdash_o A \vee B$)
- They then investigated fragments of intuitionistic logics for which it is complete, calling them “**abstract logic programming languages**”.
- The largest such fragment they identified, “**hereditarily Harrop logic**”, is in some sense maximal (via interpolation) [Har94].
- Not a complete specification: what to do with *atomic goals*?

A natural goal-directed choice is **backward-chaining** – cf. Coq’s *apply* tactic

These ideas led to the development of λ -PROLOG.

2 Proof Search in Proof Theory

Proof Theory and Proof Search

Proof search in proof theory is an active field, with surprisingly little interaction with logic programming.

A common approach: invent clever (often complicated) sequent calculi with structural properties that facilitate proof search (eg: contraction-free [Dyc92], permutation-free [DP96], focused [How98] [Sim12]).

- Unfortunately, all that cleverness tends to make them difficult to reason about
- Being “far from” natural deduction, it can be non-trivial to show equivalence, cut-admissibility; to add proof terms, etc.
- But this work need be done only once, so this approach is certainly workable (cf. Coq’s *firstorder* tactic)

Proof Theory and Proof Search

Proof search in proof theory is an active field, with surprisingly little interaction with logic programming.

A common approach: invent clever (often complicated) sequent calculi with structural properties that facilitate proof search (eg: contraction-free [Dyc92], permutation-free [DP96], focused [How98] [Sim12]).

- Unfortunately, all that cleverness tends to make them difficult to reason about
- Being “far from” natural deduction, it can be non-trivial to show equivalence, cut-admissibility; to add proof terms, etc.
- But this work need be done only once, so this approach is certainly workable (cf. Coq’s *firstorder* tactic)

Search Strategies in Sequent Calculi

A different approach: stick with a simple sequent calculus “close to” natural deduction and add a layer of **search strategy** on top.

This approach can describe Miller’s **uniform proof**(goal-directed strategy), Simmons’ **structural focalization** (via erasure), potentially others.

We investigate this approach in a categorical setting.

Why Categorical Proof Theory?

Advantages of categorical proof theory:

- Categorical logic / type theory a very active field, with many powerful tools from category theory proper
- Allows us to choose how much syntax we care about (eg: “projections instead of variables” (open- α -equivalence) [AM89], normal proof-term equivalence ($\beta\eta$ -equivalence) [Law69] [See83], proof-term rewriting [Gha95] [Gar09])
- Structural properties can be built-in (eg: interpreting contexts as cartesian products realize *weakening* as projection and *contraction* as internal diagonal)
- Mostly does away with distinction between natural deduction and sequent calculus (N.D. derivations and S.C. proofs are just *arrows*)
- Uniform treatment of *semantics*: build a generic model in the classifying category of a logic, then models are simply functors [LS86]

3 Adjunctions in Categorical Proof Theory

Adjointness in Foundations

Each connective of first-order intuitionistic logic is determined by an adjunction (thanks Bill Lawvere! [Law69])

It turns out that this fact alone is enough to ensure many good properties of the logic:

- harmony (local soundness and completeness)
- permutation conversions of \perp , \vee , \exists
- normalizability of natural deduction derivations (thanks Dag Prawitz! [Pra65])
- sequent cut elimination
- etc. (?)

Most inference rules and properties of first-order intuitionistic natural deduction and sequent calculus arise *uniformly* from the adjoint formulations of the connectives.

So we can generate proof theory from category theory.

Adjointness in Foundations

Each connective of first-order intuitionistic logic is determined by an adjunction (thanks Bill Lawvere! [Law69])

It turns out that this fact alone is enough to ensure many good properties of the logic:

- harmony (local soundness and completeness)
- permutation conversions of \perp , \vee , \exists
- normalizability of natural deduction derivations (thanks Dag Prawitz! [Pra65])
- sequent cut elimination
- etc. (?)

Most inference rules and properties of first-order intuitionistic natural deduction and sequent calculus arise *uniformly* from the adjoint formulations of the connectives.

So we can generate proof theory from category theory.

Example Right Connective

Conjunction is
an introduction and two elimination **inference rules**:

$$\frac{A \quad B}{A \wedge B} \wedge+ \qquad \frac{A \wedge B}{A} \wedge-1 \qquad \frac{A \wedge B}{B} \wedge-2$$

which are *harmonious*:

$$\frac{\frac{\frac{\mathcal{D}_1}{A_1} \quad \frac{\mathcal{D}_2}{A_2}}{A_1 \wedge A_2} \wedge+}{A_i} \wedge-i \quad \Rightarrow_{\wedge R} \quad \frac{\mathcal{D}_i}{A_i}$$

$$\frac{\varepsilon}{A \wedge B} \Rightarrow_{\wedge E} \quad \frac{\frac{\frac{\varepsilon}{A \wedge B}}{A} \wedge-1 \quad \frac{\frac{\varepsilon}{A \wedge B}}{B} \wedge-2}{A \wedge B} \wedge+$$

Example Right Connective

Conjunction is

a cartesian product, i.e. the **right adjoint** to a diagonal functor ($\Delta \dashv \wedge$):

PROP :

$$\begin{array}{ccc}
 & \Gamma \wedge \Gamma & \\
 \Delta(\Gamma) \uparrow & \searrow^{\mathcal{D}_1 \wedge \mathcal{D}_2} & \\
 \Gamma & \xrightarrow{\langle \mathcal{D}_1, \mathcal{D}_2 \rangle} & A \wedge B
 \end{array}$$

PROP \times PROP :

$$\begin{array}{ccc}
 \Delta\Gamma & \xrightarrow{(\mathcal{D}_1, \mathcal{D}_2)} & (A, B) \\
 & \searrow^{\Delta\langle \mathcal{D}_1, \mathcal{D}_2 \rangle} & \uparrow^{(fst, snd)(A, B)} \\
 & & \Delta(A \wedge B)
 \end{array}$$

Deriving the Introduction Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{\Delta\Gamma}{\mathcal{D}}}{(A, B)} \xrightarrow{-^b} \frac{\Gamma \quad \frac{\frac{\overline{\Delta\Gamma}}{\mathcal{D}}}{(A, B)}}{\wedge(A, B)}$$

- The natural isomorphism $-^b$ is *interchangeable* for $\wedge+$
- By being a natural isomorphism, it is automatically **invertible**
- The rule determines an **arrow** from the **major premise** to the **conclusion** of the rule in one category (PROP) from an arrow (derivation) determined by the minor premise in another category ($\text{PROP} \times \text{PROP}$).

Deriving the Introduction Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{\Gamma}{\mathcal{D}_1} \quad \frac{\Gamma}{\mathcal{D}_2}}{A \quad B} \xrightarrow{-^b} \frac{\Gamma \quad \frac{\frac{\bar{\Gamma}}{\mathcal{D}_1} \quad \frac{\bar{\Gamma}}{\mathcal{D}_2}}{A \quad B}}{A \wedge B} \wedge+^*$$

- The natural isomorphism $-^b$ is *interchangeable* for $\wedge+$
- By being a natural isomorphism, it is automatically **invertible**
- The rule determines an **arrow** from the **major premise** to the **conclusion** of the rule in one category (PROP) from an arrow (derivation) determined by the minor premise in another category ($\text{PROP} \times \text{PROP}$).

Deriving the Introduction Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{\Gamma}{\mathcal{D}_1} \quad \frac{\Gamma}{\mathcal{D}_2}}{A \quad B} \xrightarrow{-^b} \frac{\Gamma \quad \frac{\frac{\bar{\Gamma}}{\mathcal{D}_1} \quad \frac{\bar{\Gamma}}{\mathcal{D}_2}}{A \quad B}}{A \wedge B} \wedge+^*$$

- The natural isomorphism $-^b$ is *interchangeable* for $\wedge+$
- By being a natural isomorphism, it is automatically **invertible**
- The rule determines an **arrow** from the **major premise** to the **conclusion** of the rule in one category (PROP) from an arrow (derivation) determined by the minor premise in another category ($\text{PROP} \times \text{PROP}$).

Deriving the Elimination Rule

The counit of the adjunction ε is the ordered pair of projections in $\text{PROP} \times \text{PROP}$:

$$A \quad B \quad \xrightarrow{\varepsilon} \quad \frac{A \wedge B}{A} \quad fst \quad \frac{A \wedge B}{B} \quad snd$$

This is exactly the ordered pair of elimination rules for \wedge .

Deriving the Elimination Rule

The counit of the adjunction ε is the ordered pair of projections in $\text{PROP} \times \text{PROP}$:

$$A \quad B \quad \xrightarrow{\varepsilon} \quad \frac{A \wedge B}{A} \wedge^{-1} \quad \frac{A \wedge B}{B} \wedge^{-2}$$

This is exactly the ordered pair of elimination rules for \wedge .

Local Soundness

We can now see how the local reduction (in $\text{PROP} \times \text{PROP}$!):

$$\frac{\frac{\frac{\Gamma}{\mathcal{D}_1}}{A} \quad \frac{\frac{\Gamma}{\mathcal{D}_2}}{B}}{A \wedge B} \wedge+^*}{\frac{A \wedge B}{A} \wedge-1} \quad \frac{\frac{\frac{\Gamma}{\mathcal{D}_1}}{A} \quad \frac{\frac{\Gamma}{\mathcal{D}_2}}{B}}{A \wedge B} \wedge+^*}{\frac{A \wedge B}{B} \wedge-2} \quad \Rightarrow_{\wedge R} \quad \frac{\Gamma}{\mathcal{D}_1} \quad \frac{\Gamma}{\mathcal{D}_2}$$

is natural deduction notation for:

$$\Delta((\mathcal{D}_1, \mathcal{D}_2)^b) \cdot \varepsilon(A, B) \stackrel{\beta}{=} (\mathcal{D}_1, \mathcal{D}_2)$$

This is the universal property of the counit.

Local Soundness

We can now see how the local reduction (in $\text{PROP} \times \text{PROP}$!):

$$\frac{\frac{\frac{\Gamma}{\mathcal{D}_1}}{A} \quad \frac{\frac{\Gamma}{\mathcal{D}_2}}{B}}{A \wedge B} \wedge+^* \quad \frac{\frac{\Gamma}{\mathcal{D}_1}}{A} \quad \frac{\frac{\Gamma}{\mathcal{D}_2}}{B}}{B} \wedge-2}{\frac{\Gamma}{A}} \wedge-1 \quad \Rightarrow_{\wedge R} \quad \frac{\Gamma}{\mathcal{D}_1} \quad \frac{\Gamma}{\mathcal{D}_2}}{A} \quad \frac{\Gamma}{B}$$

is natural deduction notation for:

$$\Delta((\mathcal{D}_1, \mathcal{D}_2)^b) \cdot \varepsilon(A, B) \stackrel{\beta}{=} (\mathcal{D}_1, \mathcal{D}_2)$$

This is the universal property of the counit.

Local Completeness

Similarly, the local expansion:

$$\frac{\frac{\Gamma}{\varepsilon}}{A \wedge B} \Rightarrow_{\wedge E} \frac{\frac{\frac{\frac{\Gamma}{\varepsilon}}{A \wedge B}}{A} \wedge^{-1} \quad \frac{\frac{\frac{\Gamma}{\varepsilon}}{A \wedge B}}{B} \wedge^{-2}}{A \wedge B} \wedge^{+*}}$$

is natural deduction notation for:

$$\mathcal{E} = (\mathcal{E}^\#)^b \stackrel{(\beta^{-1})^b}{=} (\Delta \mathcal{E} \cdot \varepsilon(A, B))^b$$

Natural Deduction for Right Connectives

Pleasingly, the same relationship holds for all the connectives defined by right adjoint functors (\top , \wedge , \supset , \forall^1). We call them **right connectives**.

- introduction rule corresponds to taking the adjoint complement $-^b$
- elimination rule corresponds to composing the component of the counit ε
- local reduction corresponds to taking the shortcut β
- local expansion corresponds to conjugating $-^\#$ with β^{-1}

¹almost

Natural Deduction for Left Connectives

The situation for connectives defined by left adjoint functors (\perp , \vee , \exists), **left connectives**, is nearly dual but there is a complication:

Intuitionistic derivations are *asymmetrical*: they must have a single conclusion but may have multiple premisses.

Because the natural bijections of left connectives are determined by “arrows out of” rather than “arrows into” them, it is necessary to ensure that their elimination rules are **compatible with contexts**.

This requires satisfying the *distributive law* and *Frobenius reciprocity*:

$$\Gamma, A \vee B \equiv (\Gamma \wedge A) \vee (\Gamma \wedge B) \qquad \Gamma, \exists x . A \equiv \exists x . \Gamma \wedge A$$

Categorically, we get this for free in BCCC (category of (small) bicartesian closed categories and functors) – Thanks Nobuo Yoneda!

Natural Deduction for Left Connectives

The situation for connectives defined by left adjoint functors (\perp , \vee , \exists), **left connectives**, is nearly dual but there is a complication:

Intuitionistic derivations are *asymmetrical*: they must have a single conclusion but may have multiple premisses.

Because the natural bijections of left connectives are determined by “arrows out of” rather than “arrows into” them, it is necessary to ensure that their elimination rules are **compatible with contexts**.

This requires satisfying the *distributive law* and *Frobenius reciprocity*:

$$\Gamma, A \vee B \equiv (\Gamma \wedge A) \vee (\Gamma \wedge B) \qquad \Gamma, \exists x . A \equiv \exists x . \Gamma \wedge A$$

Categorically, we get this for free in BCCC (category of (small) bicartesian closed categories and functors) – Thanks Nobuo Yoneda!

Natural Deduction for Left Connectives

The situation for connectives defined by left adjoint functors (\perp , \vee , \exists), **left connectives**, is nearly dual but there is a complication:

Intuitionistic derivations are *asymmetrical*: they must have a single conclusion but may have multiple premisses.

Because the natural bijections of left connectives are determined by “arrows out of” rather than “arrows into” them, it is necessary to ensure that their elimination rules are **compatible with contexts**.

This requires satisfying the *distributive law* and *Frobenius reciprocity*:

$$\Gamma, A \vee B \equiv (\Gamma \wedge A) \vee (\Gamma \wedge B) \qquad \Gamma, \exists x . A \equiv \exists x . \Gamma \wedge A$$

Categorically, we get this for free in BCCC (category of (small) bicartesian closed categories and functors) – Thanks Nobuo Yoneda!

Natural Deduction for Left Connectives

The situation for connectives defined by left adjoint functors (\perp , \vee , \exists), **left connectives**, is nearly dual but there is a complication:

Intuitionistic derivations are *asymmetrical*: they must have a single conclusion but may have multiple premisses.

Because the natural bijections of left connectives are determined by “arrows out of” rather than “arrows into” them, it is necessary to ensure that their elimination rules are **compatible with contexts**.

This requires satisfying the *distributive law* and *Frobenius reciprocity*:

$$\Gamma, A \vee B \quad \equiv \quad (\Gamma \wedge A) \vee (\Gamma \wedge B) \qquad \Gamma, \exists x . A \quad \equiv \quad \exists x . \Gamma \wedge A$$

Categorically, we get this for free in BCCC (category of (small) bicartesian closed categories and functors) – Thanks Nobuo Yoneda!

Example Left Connective

Existential quantification is
a pair of **inference rules**:

$$\frac{t : X \quad A[x \mapsto t]}{\exists x : X . A} \exists+$$

$$\frac{\exists x : X . A \quad \frac{[e : X] \quad [A[x \mapsto e]]}{\mathcal{D}}}{B} \exists-\dagger$$

\dagger e may not occur outside of \mathcal{D} or in any open premise

which satisfy:

- local soundness
- local completeness
- permutation conversion

Example Left Connective

Existential quantification is

in a *hyperdoctrine*, the **left adjoint** to a typing-context weakening ($\exists \dot{x} \dashv \dot{x}^*$):

$$\text{PROP}(\Phi, x : X) : \begin{array}{ccc} & \dot{x}^*(\exists x : X . A) & \\ & \uparrow \text{ins}_x(A) & \searrow \dot{x}^*(\text{ind}_x \mathcal{D}) \\ A & \xrightarrow{\mathcal{D}} & \dot{x}^* B \\ & \beta' \Downarrow & \end{array}$$

$$\text{PROP}(\Phi) : \begin{array}{ccc} \exists x : X . A & \xrightarrow{\text{ind}_x \mathcal{D}} & B \\ & \searrow \exists x : X . \mathcal{D} & \uparrow \text{vac}_x(B) \\ & & \exists x : X . \dot{x}^* B \end{array}$$

Deriving the Elimination Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{A}{\mathcal{D}}}{\dot{x}^*(B)} \quad \xrightarrow{-\#} \quad \frac{\frac{\frac{\overline{A}}{\mathcal{D}}}{\dot{x}^*(B)}}{\exists \dot{x}(A)}}{B}$$

But this is *not* interchangeable for $\exists-$. The problem is that there is in general an ambient logical context, Γ , that may be used in both branches. Given:

$$\mathcal{E} : \text{PROP}(\Phi) (\Gamma \rightarrow \exists x : X . A) \quad \mathcal{D} : \text{PROP}(\Phi, x : X) (\dot{x}^* \Gamma \wedge A \rightarrow \dot{x}^*(B))$$

we get the rule we want if we can fill the gap f :

$$\Gamma \xrightarrow{\langle \text{id}, \mathcal{E} \rangle} \Gamma \wedge \exists x : X . A \dashrightarrow \frac{f}{\exists x : X . (\dot{x}^* \Gamma \wedge A)} \xrightarrow{\mathcal{D}^\#} B$$

This is just Frobenius reciprocity (context distributivity for \exists).

Deriving the Elimination Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{A}{\mathcal{D}}}{B} \xrightarrow{-\#} \frac{\frac{\frac{\overline{A}}{\mathcal{D}}}{B}}{\exists x : X . A} \exists -^*$$

But this is *not* interchangeable for $\exists -$. The problem is that there is in general an ambient logical context, Γ , that may be used in both branches. Given:

$$\mathcal{E} : \text{PROP}(\Phi) (\Gamma \rightarrow \exists x : X . A) \quad \mathcal{D} : \text{PROP}(\Phi, x : X) (\dot{x}^* \Gamma \wedge A \rightarrow \dot{x}^*(B))$$

we get the rule we want if we can fill the gap f :

$$\Gamma \xrightarrow{\langle \text{id}, \mathcal{E} \rangle} \Gamma \wedge \exists x : X . A \dashrightarrow \frac{f}{\exists x : X . (\dot{x}^* \Gamma \wedge A)} \dashrightarrow \frac{\mathcal{D}^\#}{B}$$

This is just Frobenius reciprocity (context distributivity for \exists).

Deriving the Elimination Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{A}{\mathcal{D}}}{B} \xrightarrow{-\#} \frac{\frac{\frac{\overline{A}}{\mathcal{D}}}{B}}{\exists x : X . A} \exists -^*$$

But this is *not* interchangeable for $\exists -$. The problem is that there is in general an ambient logical context, Γ , that may be used in both branches. Given:

$$\mathcal{E} : \text{PROP}(\Phi) (\Gamma \rightarrow \exists x : X . A) \quad \mathcal{D} : \text{PROP}(\Phi, x : X) (\dot{x}^* \Gamma \wedge A \rightarrow \dot{x}^*(B))$$

we get the rule we want if we can fill the gap f :

$$\Gamma \xrightarrow{\langle \text{id}, \mathcal{E} \rangle} \Gamma \wedge \exists x : X . A \dashrightarrow \frac{f}{\exists x : X . (\dot{x}^* \Gamma \wedge A)} \dashrightarrow \frac{\mathcal{D}^\#}{B}$$

This is just Frobenius reciprocity (context distributivity for \exists).

Deriving the Elimination Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{A}{\mathcal{D}}}{B} \xrightarrow{-\#} \frac{\frac{\frac{\overline{A}}{\mathcal{D}}}{B}}{\exists x : X . A} \exists -^*$$

But this is *not* interchangeable for $\exists -$. The problem is that there is in general an ambient logical context, Γ , that may be used in both branches. Given:

$$\mathcal{E} : \text{PROP}(\Phi) (\Gamma \rightarrow \exists x : X . A) \quad \mathcal{D} : \text{PROP}(\Phi, x : X) (\dot{x}^* \Gamma \wedge A \rightarrow \dot{x}^*(B))$$

we get the rule we want if we can fill the gap f :

$$\Gamma \xrightarrow{\langle \text{id}, \mathcal{E} \rangle} \Gamma \wedge \exists x : X . A \text{ --- } \frac{f}{\text{---}} \text{ --- } \exists x : X . (\dot{x}^* \Gamma \wedge A) \xrightarrow{\mathcal{D}^\#} B$$

This is just Frobenius reciprocity (context distributivity for \exists).

Deriving the Elimination Rule

In the syntax of derivations, this tells us that there is a natural bijection:

$$\frac{\frac{A}{\mathcal{D}}}{B} \xrightarrow{-\#} \frac{\frac{\frac{\overline{A}}{\mathcal{D}}}{B}}{\exists x : X . A} \exists -^*$$

But this is *not* interchangeable for $\exists -$. The problem is that there is in general an ambient logical context, Γ , that may be used in both branches. Given:

$$\mathcal{E} : \text{PROP}(\Phi) (\Gamma \rightarrow \exists x : X . A) \quad \mathcal{D} : \text{PROP}(\Phi, x : X) (\dot{x}^* \Gamma \wedge A \rightarrow \dot{x}^*(B))$$

we get the rule we want if we can fill the gap f :

$$\Gamma \xrightarrow{\langle \text{id}, \mathcal{E} \rangle} \Gamma \wedge \exists x : X . A \text{ --- } \frac{f}{\text{---}} \text{ --- } \exists x : X . (\dot{x}^* \Gamma \wedge A) \xrightarrow{\mathcal{D}^\#} B$$

This is just Frobenius reciprocity (context distributivity for \exists).

Deriving the Introduction Rule

The unit of the adjunction η :

$$A \xrightarrow{\eta} \frac{A}{\dot{x}^*(\exists x : X . A)} \exists +^*$$

Also not interchangeable for $\exists +$: lives in the wrong category ($\text{PROP}(\Phi, x : X)$).

But it can be reindexed along any term-in-context $\Phi \vdash t : X$

$$\begin{array}{ccc}
 \exists x : X . A & & \dot{x}^*(\exists x : X . A) \\
 & & \uparrow \eta(A) \\
 & & A \\
 \\
 \exists x : X . A & & \exists x : X . A \\
 & & \uparrow t^*(\eta(A)) \\
 & & A[x \mapsto t]
 \end{array}$$

$$\begin{array}{ccc}
 \Phi & \xleftarrow{\dot{x}} & \Phi, x : X & \xleftarrow{t} & \Phi \\
 & \searrow \text{id} & & & \swarrow
 \end{array}$$

Deriving the Introduction Rule

The unit of the adjunction η :

$$A \quad \xrightarrow{\eta} \quad \frac{A}{\dot{x}^*(\exists x : X . A)} \quad \exists +^*$$

Also not interchangeable for $\exists +$: lives in the wrong category ($\text{PROP}(\Phi, x : X)$).

But it can be reindexed along any term-in-context $\Phi \vdash t : X$

$$\begin{array}{ccccc}
 \exists x : X . A & & \dot{x}^*(\exists x : X . A) & & \exists x : X . A \\
 & & \uparrow \eta(A) & & \uparrow t^*(\eta(A)) \\
 & & A & & A[x \mapsto t] \\
 \\
 \Phi & \xleftarrow{\dot{x}} & \Phi, x : X & \xleftarrow{t} & \Phi \\
 & \searrow \text{id} & & \swarrow & \\
 & & & &
 \end{array}$$

Deriving the Permutation Conversion

Local reductions and expansions are pretty straightforward.

We point out the permutation conversion required for the proof of normalizability:

$$\frac{\frac{\frac{\frac{[A]}{\mathcal{D}}}{\dot{x}^*B}}{\exists x : X . A}}{\frac{B}{\mathcal{E}}}{C}}{\exists -} \quad \Rightarrow_{\exists P} \quad \frac{\frac{\frac{\frac{[A]}{\mathcal{D}}}{\dot{x}^*B}}{\dot{x}^*\mathcal{E}}}{\exists x : X . A}}{C}}{\exists -}$$

Categorically, this is the forward direction of:

$$D^\# \cdot \mathcal{E} = (D \cdot \dot{x}^*(\mathcal{E}))^\#$$

which is a direct consequence of the naturality of the bijection of the adjunction.

Deriving the Permutation Conversion

Local reductions and expansions are pretty straightforward.

We point out the permutation conversion required for the proof of normalizability:

$$\frac{\frac{\frac{\frac{[A]}{D}}{\dot{x}^*B}}{\exists x : X . A}}{\frac{B}{\frac{\varepsilon}{C}}} \exists-}{\Rightarrow_{\exists P}} \frac{\frac{\frac{\frac{[A]}{D}}{\dot{x}^*B}}{\dot{x}^*\varepsilon}}{\exists x : X . A}}{C} \exists-$$

Categorically, this is the forward direction of:

$$D^\sharp \cdot \varepsilon = (D \cdot \dot{x}^*(\varepsilon))^\sharp$$

which is a direct consequence of the naturality of the bijection of the adjunction.

Categorical Sequent Calculus

These ideas extend directly to sequent calculi “close to” natural deduction.

We have used this to give **categorical operational interpretation** to Miller’s connectives as *search instructions*.

Points to note:

- Right rules of right connectives and left rules of left connectives (“**eigenrules**”) take adjoint complements, (modulo left context distribution) so are invertible
- Most “**anderrules**” compose a (co)unit, those for the quantifiers allow us to delay the choice of instantiating term (cf: Coq’s “e”-tactics)
- Left rule for \supset a bit different (incorporates a built-in cut), restricting this leads to various (partial) proof search strategies (eg: backward and forward chaining)
- Free hyperdoctrine (indexed bicartesian closed category) provides a categorical generic model for proof search

Categorical Sequent Calculus

These ideas extend directly to sequent calculi “close to” natural deduction.

We have used this to give **categorical operational interpretation** to Miller’s connectives as *search instructions*.

Points to note:

- Right rules of right connectives and left rules of left connectives (“**eigenrules**”) take adjoint complements, (modulo left context distribution) so are invertible
- Most “**anderrules**” compose a (co)unit, those for the quantifiers allow us to delay the choice of instantiating term (cf: Coq’s “e”-tactics)
- Left rule for \supset a bit different (incorporates a built-in cut), restricting this leads to various (partial) proof search strategies (eg: backward and forward chaining)
- Free hyperdoctrine (indexed bicartesian closed category) provides a categorical generic model for proof search

4 Connective Chirality and Search Strategy

Constructivity of Intuitionistic Logic

Can be understood to mean:

Disjunction Property

$$\vdash A \vee B \quad \Leftrightarrow \quad \vdash A \quad \text{or} \quad \vdash B$$

Existence Property

$$\vdash \exists x : X . A \quad \Leftrightarrow \quad \vdash A [x \mapsto t] \quad \text{for some } t : X$$

By analogy, we add:

Falsity Property

$$\vdash \perp \quad \Leftrightarrow \quad \text{never}$$

which simply expresses *consistency*.

Constructivity of Intuitionistic Logic

Can be understood to mean:

Disjunction Property

$$\vdash A \vee B \quad \Leftrightarrow \quad \vdash A \quad \text{or} \quad \vdash B$$

Existence Property

$$\vdash \exists x : X . A \quad \Leftrightarrow \quad \vdash A [x \mapsto t] \quad \text{for some } t : X$$

By analogy, we add:

Falsity Property

$$\vdash \perp \quad \Leftrightarrow \quad \text{never}$$

which simply expresses *consistency*.

Constructivity of Intuitionistic Logic

In the '60's, Kleene and Harrop noticed:

Strong Disjunction Property: If Γ contains no strictly positive disjunction then:

$$\Gamma \vdash A \vee B \quad \Leftrightarrow \quad \Gamma \vdash A \quad \text{or} \quad \Gamma \vdash B$$

Strong Existence Property: If Γ contains no strictly positive disjunction or existential then:

$$\Gamma \vdash \exists x : X . A \quad \Leftrightarrow \quad \Gamma \vdash A [x \mapsto t] \quad \text{for some } t : X$$

Continuing the analogy:

Strong Falsity Property: If Γ contains no strictly positive \perp (negation) then:

$$\Gamma \vdash \perp \quad \Leftrightarrow \quad \text{never}$$

Constructivity of Intuitionistic Logic

In the '60's, Kleene and Harrop noticed:

Strong Disjunction Property: If Γ contains no strictly positive disjunction then:

$$\Gamma \vdash A \vee B \quad \Leftrightarrow \quad \Gamma \vdash A \quad \text{or} \quad \Gamma \vdash B$$

Strong Existence Property: If Γ contains no strictly positive disjunction or existential then:

$$\Gamma \vdash \exists x : X . A \quad \Leftrightarrow \quad \Gamma \vdash A [x \mapsto t] \quad \text{for some } t : X$$

Continuing the analogy:

Strong Falsity Property: If Γ contains no strictly positive \perp (negation) then:

$$\Gamma \vdash \perp \quad \Leftrightarrow \quad \text{never}$$

Constructive Sequents

This gives a sufficient condition for the *invertibility* of right rules of left connectives.

Recall that right rules of right connectives are *always* invertible by the natural bijections of their adjunctions.

By restricting left connectives to occur only positively on the right and only negatively on the left of a sequent we have a system where all right rules are invertible. We call these “**constructive sequents**”.

This validates the **completeness** of **goal directed search** for constructive sequents.

Constructive Sequents

This gives a sufficient condition for the *invertibility* of right rules of left connectives.

Recall that right rules of right connectives are *always* invertible by the natural bijections of their adjunctions.

By restricting left connectives to occur only positively on the right and only negatively on the left of a sequent we have a system where all right rules are invertible. We call these “**constructive sequents**”.

This validates the **completeness** of **goal directed search** for constructive sequents.

Hereditarily Harrop Logic

Recursive grammar for constructive sequents

Let \mathcal{D} and \mathcal{G} be the sets of **constructive antecedents** (“program formulae”) and **constructive succedents** (“goal formulae”).

They may be defined by the recursive grammars:

$$\begin{aligned} \mathcal{D} & ::= P \mid \top \mid D \wedge D \mid \forall x : X . D \mid G \supset D \\ \mathcal{G} & ::= P \mid \top \mid \perp \mid G \wedge G \mid G \vee G \mid \forall x : X . G \mid \exists x : X . G \mid D \supset G \end{aligned}$$

where P is the class of atomic propositions.

This is essentially the same as **hereditarily Harrop** program and goal formula, which don't contain \top and \perp , respectively (the enrichment is of little practical use).

Hereditarily Harrop Logic

Recursive grammar for constructive sequents

Let \mathcal{D} and \mathcal{G} be the sets of **constructive antecedents** (“program formulae”) and **constructive succedents** (“goal formulae”).

They may be defined by the recursive grammars:

$$\mathcal{D} ::= P \mid \top \mid D \wedge D \mid \forall x : X . D \mid G \supset D$$

$$\mathcal{G} ::= P \mid \top \mid \perp \mid G \wedge G \mid G \vee G \mid \forall x : X . G \mid \exists x : X . G \mid D \supset G$$

where P is the class of atomic propositions.

This is essentially the same as **hereditarily Harrop** program and goal formula, which don't contain \top and \perp , respectively (the enrichment is of little practical use).

Goal Directed Search for Atoms

- Recall only right propositions may occur positively in constructive contexts (antecedents)
- Positive occurrences of right connectives can be rewritten by the natural bijection of their adjunction
- In particular, constructive contexts can be rewritten with the propositions in *strictly positive* position all atoms
- This allows a strategy of goal directed search for atoms

Backward Chaining

$$\frac{
 \frac{
 \frac{
 \frac{
 \frac{
 \frac{
 \Gamma \Rightarrow G' \quad \frac{(P'\sigma = P\sigma)}{\Gamma, P' \Rightarrow P} \textit{init}
 }{\Gamma, G' \supset P' \Rightarrow P} \supset L
 }{x_n : X_n} \forall L
 }{\vdots} \forall L
 }{x_1 : X_1} \forall L
 }{\Gamma, \forall \vec{x} : \vec{X}. G' \supset P' \Rightarrow P} cL
 }{\Gamma \Rightarrow P} *R
 }{\vdots}
 }{\Gamma \Rightarrow P}$$

Categorically, this corresponds to pre-composing an arrow named by a program clause with a derivation of an instance of the goal.

References



Andrea Asperti and Simone Martini. "Projections Instead of Variables: a category theoretic interpretation of logic programs". In: *International Conference for Logic Programming*. MIT Press, 1989.



Roy Dyckhoff. "Contraction-Free Sequent Calculi for Intuitionistic Logic". In: *Journal of Symbolic Logic* 57.3 (1992), pp. 795–807.



Roy Dyckhoff and Luís Pinto. "Uniform Proofs and Natural Deductions". In: *International Conference on Automated Deduction*. 1994.



Roy Dyckhoff and Luís Pinto. *A Permutation-Free Sequent Calculus for Intuitionistic Logic*. Tech. rep. St. Andrews University, 1996.



Richard Garner. "Two-Dimensional Models of Type Theory". In: *Mathematical Structures in Computer Science* 19.4 (2009), 687–736. URL: <http://arxiv.org/abs/0808.2122v2>.



Neil Ghani. "Adjoint Rewriting". PhD thesis. University of Edinburgh, 1995.



James Harland. "A Proof-theoretic Analysis of Goal-directed Provability". In: *Journal of Logic and Computation* 4.1 (1994), pp. 69–88.



Jacob Howe. "Proof Search Issues in Some Non-Classical Logics". PhD thesis. University of St. Andrews, 1998.



Joachim Lambek and Philip Scott. *Introduction to Higher Order Categorical Logic*. Cambridge University Press, 1986.



F. William Lawvere. "Adjointness in Foundations". In: *Dialectica* 23 (1969).



Dale Miller. "A Logical Analysis of Modules in Logic Programming". In: *Journal of Logic Programming* 6 (1989), pp. 79–108.



Dale Miller et al. "Uniform Proofs as a Foundation for Logic Programming". In: *Annals of Pure and Applied Logic* 51 (1991).



Dag Prawitz. *Natural Deduction: A Proof-Theoretical Study*. Stockholm Studies in Philosophy 3. Almqvist and Wiksell, 1965.



Robert Seely. "Hyperdoctrines, Natural Deduction and the Beck Condition". In: *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik* (1983).



Robert J. Simmons. "Structural Focalization". In: (2012).