Towards Proof Planning for $\mathcal{M}_\omega^+$

Carsten Schürmann
Yale University

Serge Autexier
German Research Center for Artificial Intelligence (DFKI)

LFM 2002

Motivation

Specification

- Operational Semantics
- Static Semantics (Typing)
- Derivability

Deductive System

- Judgments

$$e \hookrightarrow v \qquad \Gamma \vdash e : \tau \qquad \Gamma \vdash A$$

- Inference rules

$$\frac{e \hookrightarrow v}{s\ e \hookrightarrow s\ v} \qquad \frac{\Gamma \vdash e_1 : \tau_2 \rightarrow \tau_1 \quad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash e_1\ e_2 : \tau_1} \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$$

Motivation

Theorems we are interested in

- Type preservation

  For all $e \hookrightarrow v$ and $\cdot \vdash e : \tau$ then $\cdot \vdash v : \tau$.

- Embedding of one type system in another

  For all $\Gamma \vdash e : \tau$ then $[\![\Gamma]\!] \vdash [\![e]\!] : [\![\tau]\!]$.

- Consistency

  For all $\Gamma \vdash A$ and $\Gamma, A \vdash B$ then $\Gamma \vdash B$

Proofs

- By structural induction
- Using inversion

Challenges

Automation

- Combinatorical explosion

  Which rule to apply next?

- Non deterministic choice

  Which assumption to consider cases over next?

- Level of abstraction

  Which operators to offer?

Other proof techniques

- Logical relations
- Proof libraries

Our Contribution

Proof planning calculus $\mathcal{P}_\omega^+$

- Recognizes unpromising states.
- Provides proof search guidance.
- Gives logical explanation to proof plans.
- Works with meta logic $\mathcal{M}_\omega^+$. [Schürmann]
- Defined for Logical Framework LF. [Harper et al.]

Intuition

- Approximates information contained in dependent types.
- Supports reasoning natural deduction style.
- Proof plans are natural deduction derivations.

Overview

Extended Example

Formal Treatment

- Logical Framework LF
- Meta logic $\mathcal{M}_\omega^+$
- Proof term calculus $\mathcal{P}_\omega^+$
- Approximation from $\mathcal{M}_\omega^+$ to $\mathcal{P}_\omega^+$
- Meta theory of $\mathcal{P}_\omega^+$

Formalization: Type preservation

Judgments as propositions

- Substitution lemma:

$$\forall \Gamma . \forall \Gamma' . \forall x . \forall e . \forall e' . \forall e'' . \forall \tau . \forall \tau' . \mathsf{append}(\Gamma, (x, \tau'), \Gamma') \\ \wedge \;\; \mathsf{of}(\Gamma', e, \tau') \wedge \mathsf{of}(\Gamma, v, \tau') \wedge \mathsf{subst}(e, x, v, e'') \supset \mathsf{of}(\Gamma, e'', \tau)$$

- Inductive arguments require induction principles
- Indirection: Logic of propositions

Judgments as types

- Powerful meta logical framework LF+$\mathcal{M}_\omega^+$
- Higher-order representation technique
- Implicit treatment of substitution lemmas

Example: Mini-ML

Terms:

$$e ::= x \mid \mathbf{lam}\ x.e \mid \mathbf{app}\ e_1\ e_2 \mid \mathbf{fix}\ x.e$$

Some typing rules: $\Gamma \vdash e : \tau$

$$\frac{\Gamma \vdash e_1 : \tau_2 \to \tau_1 \quad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash \mathbf{app}\ e_1\ e_2 : \tau_1}\ \text{of\_app}$$

$$\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \mathbf{lam}\ x.e : \tau_1 \to \tau_2}\ \text{of\_lam} \qquad \frac{\Gamma, x : \tau \vdash e : \tau}{\Gamma \vdash \mathbf{fix}\ x.e : \tau}\ \text{of\_fix}$$

Some evaluation rules: $e \hookrightarrow v$

$$\frac{e_1 \hookrightarrow \mathbf{lam}\ x.e_1' \quad e_2 \hookrightarrow v_2 \quad [v_2/x]e_1' \hookrightarrow v}{\mathbf{app}\ e_1\ e_2 \hookrightarrow v}\ \text{ev\_app}$$

$$\frac{}{\mathbf{lam}\ x.e \hookrightarrow \mathbf{lam}\ x.e}\ \text{ev\_lam} \qquad \frac{[\mathbf{fix}\ x.e/x]e \hookrightarrow v}{\mathbf{fix}\ x.e \hookrightarrow v}\ \text{ev\_fix}$$

Example: Mini-ML (cont'd)

```
of : exp → tp → type.
of_lam : (Πx:exp. of x T₁ → of (E x) T₂)
          → of (lam E) (arrow T₁ T₂).
of_app : of E₂ T₂ → of E₁ (arrow T₂ T₁)
          → of (app E₁ E₂) T₁.
of_fix : (Πx:exp. of x T → of (E x) T)
          → of (fix E) T.

eval : exp → exp → type.
ev_lam : eval (lam E) (lam E).
ev_app : eval (E₁' V₂) V → eval E₂ V₂ → eval E₁ (lam E₁')
          → eval (app E₁ E₂) V.
ev_fix : eval (E (fix E)) V
          → eval (fix E) V.
```

Meta logic $\mathcal{M}_\omega^+$

- First-order meta logic $\mathcal{M}_\omega^+$ without propositional constants.
- Quantifiers range *directly* over canonical forms.

$$\forall E : \mathsf{exp}. \forall V : \mathsf{exp}. \forall T : \mathsf{tp} \\ \forall D : \mathsf{eval}\ E\ V. \forall P : \mathsf{of}\ E\ T. \exists Q : \mathsf{of}\ V\ T. \top$$

- Inherits from LF: Substitution lemmas.
- Worlds: "Inductive" types *with* negative occurrences.
- Proof states.
- Example:

$$E : \mathsf{exp}, V : \mathsf{exp}, T : \mathsf{tp}, D : \mathsf{eval}\ E\ V, P : \mathsf{of}\ E\ T \vdash \exists Q : \mathsf{of}\ V\ T. \top$$

- How about proof automation?

Historical Perspective

Combinatorical Explosion

- Uniform derivations [Miller et al., ...]
- Permutability [Galmiche, ...]
- Focusing [Andreoli, ...]
- Tactical theorem proving [Paulson, ...]

Level of Abstraction

- Proof planning [Bundy et al., ...]
- Rippling [Hutter et al, Bundy et al., ...]
- Admissible/derived rules of inference
- $\epsilon\delta$ proofs [Melis et al.]

Theorem Prover Operators

Splitting

- Case Analysis
- Example: D : eval E V
- Example: P : of E T
- Problem: Non-determinism

Filling

- Proof search for objects of given type
- Example: Q : of V T
- Provides witness objects
- Lemma/Induction hypotheses application
- Problem: Size of search space

Central Insight

Exploit information contained in types.

Example: $D : \text{eval } E \; V$

D contains information about E and V.
written as
$(\text{eval } E)$ and $(\text{eval } V)$

More general: If $M : \Pi x_1 : A_1. \ldots \Pi x_m : A_m. A$ then

M contains information about subterms in $A_1 \ldots A_n$, and A.

Central Idea:

- Capture this information in form of propositions.
- Omit the rest.

Example: Type Preservation

Recall: Proof state

$$E : \mathsf{exp}, V : \mathsf{exp}, T : \mathsf{tp}, D : \mathsf{eval}\ E\ V, P : \mathsf{of}\ E\ T \vdash \exists Q : \mathsf{of}\ V\ T. \top$$

Result of approximation

$$(\mathsf{eval}\ E) \wedge (\mathsf{eval}\ V), (\mathsf{of}\ E) \wedge (\mathsf{of}\ T) \vdash (\mathsf{of}\ V) \wedge (\mathsf{of}\ T)$$

Observations:

- Is there a natural deduction derivation? No!
- Problem: Proof of $(\mathsf{of}\ V)$.
- Without Splitting: Filling will fail.
- With Splitting: Filling may succeed.

Example: Type Preservation (cont'd)

Splitting D:

1. $..., P : \text{of (lam } E') \; T \vdash \exists Q : \text{of (lam } E') \; T. \top$

2. $..., D_1 : \text{eval } E_1 \; (\text{lam } E_1'), D_2 : \text{eval } E_2 \; V_2, D_3 : \text{eval } (E_1' \; V_2) \; V,$

   $P : \text{of (app } E_1 \; E_2) \; T,$

   $\mathbf{ih}_1 \in \forall t : \text{tp}. \forall u : \text{of } E_1 \; \text{t}. \exists q : \text{of (lam } E_1') \; \text{t}. \top,$

   $\mathbf{ih}_2 \in \forall t : \text{tp}. \forall u : \text{of } E_2 \; \text{t}. \exists q : \text{of } V_2 \; \text{t}. \top,$

   $\mathbf{ih}_3 \in \forall t : \text{tp}. \forall u : \text{of } (E_1' \; V_2) \; \text{t}. \exists q : \text{of } V \; \text{t}. \top,$

   $\vdash \exists Q : \text{of } V \; T. \top$

3. $..., D : \text{eval } (E' \; (\text{fix } E')) \; V, P : \text{of (fix } E') \; T,$

   $\mathbf{ih} \in \forall t : \text{tp}. \forall p : \text{of } (E' \; (\text{fix } E')) \; t. \exists q : \text{of } V \; t. \top$

   $\vdash \exists Q : \text{of } V \; T. \top$

Example: Type Preservation (cont'd)

Approximation types:

1. $(\text{of } E') \wedge (\text{of } T) \vdash (\text{of } E') \wedge (\text{of } T)$

2. $(\text{eval } E') \wedge (\text{eval } E_1'), (\text{eval } E_2) \wedge (\text{eval } V_2),$

   $(\text{eval } E_1') \wedge (\text{eval } V_2) \wedge (\text{eval } V), (\text{of } E_1) \wedge (\text{of } E_2) \wedge (\text{of } T),$

   $\forall t : \text{tp}. (\text{of } E_1) \wedge (\text{of } t) \supset (\text{of } E_1') \wedge (\text{of } t),$

   $\forall t : \text{tp}. (\text{of } E_2) \wedge (\text{of } t) \supset (\text{of } V_2) \wedge (\text{of } t),$

   $\forall t : \text{tp}. (\text{of } E_1') \wedge (\text{of } V_2) \wedge (\text{of } t) \supset (\text{of } V) \wedge (\text{of } t)$

   $\vdash (\text{of } V) \wedge (\text{of } T)$

3. $(\text{eval } E') \wedge (\text{eval } E') \wedge (\text{eval } V), (\text{of } E') \wedge (\text{of } T)$

   $\forall t : \text{tp}. (\text{of } E') \wedge (\text{of } E') \wedge (\text{of } t) \supset (\text{of } V) \wedge (\text{of } t)$

   $\vdash (\text{of } V) \wedge (\text{of } T)$

Proof Planning Calculus $\mathcal{P}_\omega^+$

All cases are provable!

$\mathcal{P}_\omega^+$ is a first-order natural deduction calculus $(\wedge, \supset, \forall)$.

Algorithm.

- Apply splitting operators.
- Compute approximation states.
- Conduct proof search in $\mathcal{P}_\omega^+$.
- Success: Find different case splits.
- Failure: Attempt to finish a proof.
  - Success: Pick new subgoal
  - Failure: Apply further splits.

Proof Planning Calculus $\mathcal{P}_\omega^+$ (cont'd)

Derived rules of inference in $\mathcal{M}_\omega^+$.

- Constants as lemmas

```
of_lam : (Πx:exp. of x T1 → of (E x) T2)
         → of (lam E) (arrow T1 T2)
```

$$(\text{of } T_1) \wedge (\text{of } E) \wedge (\text{of } T_2) \supset (\text{of } E) \wedge (\text{of } T_1) \wedge (\text{of } T_2)$$

- Parameters as lemmas

```
P' : Πx:exp. of x T1 → of (E x) T2
```

$$\forall x.(\text{of } x) \wedge (\text{of } T_1) \supset (\text{of } E) \wedge (\text{of } x) \wedge (\text{of } T_2)$$

Future work: Splits on the outcome of lemmas.

Overview

Extended Example

Formal Treatment

- **Logical Framework LF**
- **Meta logic** $\mathcal{M}_\omega^+$
- **Proof term calculus** $\mathcal{P}_\omega^+$
- **Approximation from** $\mathcal{M}_\omega^+$ **to** $\mathcal{P}_\omega^+$
- **Meta theory of** $\mathcal{P}_\omega^+$

Part 1: Logical Framework LF

Definition: [Harper Honsell Plotkin'93]

Kinds	K	$::=$	$\mathsf{type} \mid \Pi x : A. K \mid A \to K$
Types	A	$::=$	$a \mid A\ M \mid \Pi x : A_1. A_2 \mid A_1 \to A_2$
Terms	M	$::=$	$x \mid c \mid \lambda x : A. M \mid M_1\ M_2$
Signature	Σ	$::=$	$\cdot \mid \Sigma, c : A \mid \Sigma, a : K$
'Context	Γ	$::=$	$\cdot \mid \Gamma, x : A$

Theorem: Every M can be converted into canonical form.

Judgments: $\Gamma \vdash_{\Sigma;\Phi} M \Uparrow A$, $\Gamma \vdash_{\Sigma;\Phi} A \Uparrow K$, $\Gamma \vdash_{\Sigma;\Phi} K \Uparrow \mathsf{kind}$.

Remark: Blocks and worlds omitted but work. [Schürmann]

Judgment $\Psi \Vdash_{\Sigma;\Phi} \mathcal{P}$

Ψ is Γ plus induction hypotheses.

$$\frac{}{\Psi \Vdash \top}\,\text{true} \qquad \frac{\mathcal{P} \in \Psi}{\Psi \Vdash \mathcal{P}}\,\text{init}$$

$$\frac{\Psi, x : A \Vdash \mathcal{P}(x)}{\Psi \Vdash \forall x : A.\mathcal{P}(x)}\,\text{allI}^x \qquad \frac{\Psi \Vdash \forall x : A.\mathcal{P}(x) \qquad \Psi \vdash M \Uparrow A}{\Psi \Vdash \mathcal{P}(M)}\,\text{allE}$$

$$\frac{\Psi \vdash M \Uparrow A \qquad \Psi \Vdash \mathcal{P}(M)}{\Psi \Vdash \exists x : A.\mathcal{P}(x)}\,\text{exI}$$

$$\frac{\Psi \Vdash \exists x : A.\mathcal{P}(x) \qquad \Psi, x : A, \mathcal{P}(x) \Vdash \mathcal{P}'}{\Psi \Vdash \mathcal{P}'}\,\text{exE}^x$$

Part 2: Meta logic $\mathcal{M}_\omega^+$ (cont'd)

Recursion:

$$\frac{\Psi, \mathcal{P} \Vdash \mathcal{P}}{\Psi \Vdash \mathcal{P}} \text{rec}$$

- All uses of assumption $\mathcal{P}$ terminate.

Case Analysis:

$$\frac{\Psi_1 \Vdash \mathcal{P}[\sigma_1] \quad \ldots \quad \Psi_n \Vdash \mathcal{P}[\sigma_n]}{\Psi \Vdash \mathcal{P}} \text{case}$$

- $\Psi_i \Vdash \sigma_i \in \Psi$
- All cases are covered

Contexts: $\Delta ::= \cdot \mid \Delta, G$

$$\frac{G \in \Delta}{\Delta \vdash G}\ \mathsf{ax} \qquad \frac{}{\Delta \vdash \top}\ \mathsf{true}$$

$$\frac{\Delta, x \vdash G}{\Delta \vdash \forall x.\, G}\ \mathsf{allI} \qquad \frac{y \in \Delta \quad \Delta \vdash \forall x.\, G}{\Delta \vdash G[y/x]}\ \mathsf{allE}$$

$$\frac{y \in \Delta \quad \Delta \vdash G[y/x]}{\Delta \vdash \exists x.\, G}\ \mathsf{exI} \qquad \frac{\Delta \vdash \exists x.\, G \quad \Delta, y, G[y/x] \vdash G'}{\Delta \vdash G'}\ \mathsf{exE}$$

$$\frac{\Delta \vdash G_1 \quad \Delta \vdash G_2}{\Delta \vdash G_1 \wedge G_2}\ \mathsf{andI} \qquad \frac{\Delta \vdash G_1 \wedge G_2}{\Delta \vdash G_1}\ \mathsf{andE}_1 \qquad \frac{\Delta \vdash G_1 \wedge G_2}{\Delta \vdash G_2}\ \mathsf{andE}_2$$

$$\frac{\Delta, G_1 \vdash G_2}{\Delta \vdash G_1 \supset G_2}\ \mathsf{impI} \qquad \frac{\Delta \vdash G_2 \supset G_1 \quad \Delta \vdash G_2}{\Delta \vdash G_1}\ \mathsf{impE}$$

Part 4: Approximation from $\mathcal{M}_\omega^+$ to $\mathcal{P}_\omega^+$

Judgment: $\Psi \vdash A \rightsquigarrow G$, $\Psi \vdash \mathcal{P} \rightsquigarrow G$

Rules:

$$\frac{A \text{ is atomic} \qquad G \text{ corresponds to } A}{\Psi \vdash A \rightsquigarrow G}\ \text{tatom}$$

$$\frac{\Psi \vdash A_1 \rightsquigarrow G_1 \quad \Psi, x : A_1 \vdash A_2 \rightsquigarrow G_2}{\Psi \vdash \Pi x : A_1 . A_2 \rightsquigarrow G_1 \supset \forall x. G_2}\ \text{tpi}$$

$$\frac{}{\Psi \vdash \top \rightsquigarrow \top}\ \text{ttrue}$$

$$\frac{\Psi \vdash A \rightsquigarrow G_1 \quad \Psi, x : A \vdash \mathcal{P} \rightsquigarrow G_2}{\Psi \vdash \forall x : A . \mathcal{P} \rightsquigarrow G_1 \supset \forall x. G_2}\ \text{tall} \qquad \frac{\Psi \vdash A \rightsquigarrow G_1 \quad \Psi, x : A \vdash \mathcal{P} \rightsquigarrow G_2}{\Psi \vdash \exists x : A . \mathcal{P} \rightsquigarrow G_1 \wedge \exists x. G_2}\ \text{tex}$$

$$\frac{\Psi \vdash \mathcal{P}_1 \rightsquigarrow G_1 \quad \Psi \vdash \mathcal{P}_2 \rightsquigarrow G_2}{\Psi \vdash \mathcal{P}_1 \wedge \mathcal{P}_2 \rightsquigarrow G_1 \wedge G_2}\ \text{tand} \qquad \frac{\Psi \vdash \mathcal{P}_1 \rightsquigarrow G_1 \quad \Psi \vdash \mathcal{P}_2 \rightsquigarrow G_2}{\Psi \vdash \mathcal{P}_1 \supset \mathcal{P}_2 \rightsquigarrow G_1 \supset G_2}\ \text{timp}$$

Part 5: Meta Theory

In the paper

- Omitted cases.
- Approximation of contexts $\cdot \vdash \Psi \leadsto \Delta$

Theorem If $\mathcal{D} :: \Psi \vdash \mathcal{P}$

and $\mathcal{D}$ does not contain any applications to the case rule

and $\cdot \vdash \Psi \leadsto \Delta$

and $\Psi \vdash \mathcal{P} \leadsto G$

then $\Delta \vdash G$.

Proof by induction. All cases checked.

Conclusion

Proof Planning Calculus $\mathcal{P}_\omega^+$

- Clean logical foundation of proof planning.
- Scales to all of $\mathcal{M}_\omega^+$, i.e. blocks and worlds.
- Technical report is forthcoming.

Implementation

- Underway.
- Projected Date: End of the year.

Future Work

- Exploiting the proof plan for proof search.
- Interpretation of failure.

For more information

http://www.cs.yale.edu/~carsten