Coinductive Uniform Proofs

Presented by

Mr Yue Li

Researched by

Dr Henning Basold[2]

Dr Ekaterina Komendantskaya[1]

Mr Yue Li[1]

[1]Heriot-Watt University, Scotland

[2]ENS de Lyon, France

8 July 2018, Oxford, UK

Introduction

Context First-order Horn clause logic programming.

Goal Detecting non-termination by coinductive proof.

State of the Art Heuristic algorithms

1. Coinductive Logic Programming
2. Proof Relevant Corecursive Resolution

Open Problems The heuristic algorithms:

1. have limits, and
2. do not have proof theoretic foundation.

We Propose: Coinductive Uniform Proof

- A principled approach to the Goal.
- A proof theoretic foundation for the heuristic algorithms.
- Breaking through the limits of the heuristic algorithms.

Background: Fixed-point Models (aka Herbrand Models)

- Given a first-order Horn clause logic program P, in classical logic:
- The least fixed-point model contains all finite terms that can be proved to be true w.r.t P.
- The greatest fixed-point model contains all finite and infinite terms that cannot be proved to be false (i.e. either true or non-terminating) w.r.t P.

Example

- Clauses nat 0 and $\forall x.\,\text{nat x} \rightarrow \text{nat (s x)}$ intend to define the set $\mathbb{N}$ of all non-negative integers.
- A typical $n \in \mathbb{N}$ has the form $s - \cdots - s - 0$.
- The least fixed point model is $M_\mu = \{nat\ n \mid n \in \mathbb{N}\}$.
- The greatest fixed-point model is $M_\nu = M_\mu \cup \{nat\ \omega\}$
- . . . where ω is the infinite term $s - s - s - \cdots$.

Background: Coinductive Logic Programming (CoLP)

- Created by Gopal Gupta et al in 2006
- A goal succeeds if it unifies a previous goal (no occurs check)
- Being sound w.r.t. the greatest fixed-point model.

Example

- Consider the program: $\forall x.\ \text{zeros}\ x \rightarrow \text{zeros}\ [0 \mid x]$
- SLD-derivation ($\rightsquigarrow$): $\text{zeros}\ x \rightsquigarrow \text{zeros}\ x' \rightsquigarrow \cdots$

Result $[0 \mid x']/x,\ [0 \mid x'']/x',\ \cdots$

- leading towards the correct answer, but
- non-terminating

- CoLP derivation ($\rightsquigarrow$): $\text{zeros}\ x \rightsquigarrow \text{zeros}\ x'$ ✓
  - zeros x' unifies zeros x.

Result $[0 \mid x]/x$ (circular unifier, representing $[0, 0, \cdots]/x$)

- giving exactly the correct answer.

Background: Proof Relevant Corecursive Resolution (Precor)

- Created by Komendantskaya et al in 2015
- Including a heuristic to suggest a "coinductive invariant (Co-I)", plus a specially suggested calculus to prove the Co-I.
- The corresponding infinite SLD-derivation is recoverable from a Precor proof.

Background: Proof Relevant Corecursive Resolution (Precor)

Example

- Consider the program: $\forall x.\ \mathsf{paul\_loves}\ (\mathsf{dog\_of}\ x) \rightarrow \mathsf{paul\_loves}\ x$

Background: Proof Relevant Corecursive Resolution (Precor)

Example

- Consider the program: $\forall x.\ \mathsf{p}\ (\mathsf{d}\ x) \rightarrow \mathsf{p}\ x$

Background: Proof Relevant Corecursive Resolution (Precor)

Example

- Consider the program: $\forall x.\ \mathsf{p}\ (\mathsf{d}\ x) \rightarrow \mathsf{p}\ x$
- SLD-derivation ($\rightsquigarrow$): $\mathsf{p}\ x \rightsquigarrow \mathsf{p}\ (\mathsf{d}\ x) \rightsquigarrow \cdots$

Note SLD-derivation is restricted to rewriting.

☹ non-terminating, no answer.

- CoLP-derivation ($\rightsquigarrow$): $\mathsf{p}\ x \rightsquigarrow \mathsf{p}\ (\mathsf{d}\ x)$ ✓
  - $\mathsf{p}\ x$ unifies $\mathsf{p}\ (\mathsf{d}\ x)$.

Result $(d\ x)/x$ (circular unifier, denoting $[d - d - \cdots /x]$)

☺ A correct answer !

- Precor suggests a Co-I: $\forall x.\ \mathsf{p}\ x$
  - then proves the Co-I: $\forall x.\ \mathsf{p}\ x \multimap \mathsf{p}\ c \rightsquigarrow \mathsf{p}\ (\mathsf{d}\ c)$ ✓
  - $\multimap$ is introduction rule for $\forall$; $\mathsf{p}\ (\mathsf{d}\ c)$ is an instance of the Co-I.

☺ The Co-I is a correct and more general (than CoLP) answer.

☺ The pattern of the SLD-derivation is captured.

Background: Limitations of CoLP and Precor

- CoLP only works with cyclic patterns.
- Precor requires that SLD-resolution is restricted to term matching (rewriting).

Motivating Example

$$\forall xy.\ \mathsf{from}\ (s\ x)\ y \rightarrow \mathsf{from}\ x\ [x \mid y]$$

- The "from" predicate has two arguments:
- The first argument takes some number N.
- The second argument returns a stream led by N:

$$\mathsf{N},\ \mathsf{s}\ \mathsf{N},\ \mathsf{s}\ (\mathsf{s}\ \mathsf{N}),\ \mathsf{s}\ (\mathsf{s}\ (\mathsf{s}\ \mathsf{N})),\ \cdots$$

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task I: Find x and y, such that, from x y

Approach: SLD

- SLD-derivation ($\rightsquigarrow$): from x $y \rightsquigarrow$ from $(s\ x)$ $y' \rightsquigarrow \cdots$

Result $[x \mid y']/y$, $[(s\ x) \mid y'']/y'$, $\cdots$

Note Full SLD-resolution is needed, instead of just rewriting.

Note Goals *do* unify (no occurs check)

- ☺ leading towards the correct answer only for y
- ☹ non-terminating, no answer for x

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task I: Find x and y, such that, from x y

Approach: CoLP

- CoLP-derivation ($\rightsquigarrow$): from x $y \rightsquigarrow$ from $(s\ x)$ y' ✓
- from x y unifies from $(s\ x)$ y'

Result $[(s\ x)/x, [x \mid y]/y]$

☺ A correct pair of answers for both x and y!

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task I: Find x and y, such that, from x y

Approach: Precor

- ☹ N/A
- because full SLD-resolution is needed, instead of just rewriting.

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task II: Find y, such that, from $0\ y$

Approach: SLD

- SLD-derivation ($\rightsquigarrow$): from $0\ y \rightsquigarrow$ from $(s\ 0)\ y' \rightsquigarrow \cdots$

Result $[0 \mid y']/y, [(s\ 0) \mid y'']/y', \cdots$

Note Full SLD-resolution is needed, instead of just rewriting.

Note Goals do not unify (no occurs check)

- leading towards the correct answer, but
- non-terminating

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task II: Find y, such that, from 0 y

Approach: CoLP

- CoLP-derivation ($\rightsquigarrow$): from 0 $y \rightsquigarrow \cdots$
- CoLP behaves the same as SLD in this case,
- because goals do not unify (no occurs check).
- leading towards the correct answer, but
- non-terminating

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task II: Find y, such that, from 0 y

Approach: Precor

- ☹ N/A
- because full SLD-resolution is needed, instead of just rewriting.

Background: Limitations of CoLP and Precor

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

- Task I: Find x and y, such that, from x y
- Task II: Find y, such that, from 0 y

Approaches: Summary

Algorithm	Task I	Task II
SLD	☹	☹
CoLP	☺	☹
Precor	☹	☹

☺ At least one answer. ☹ No answer.

Coinductive Uniform Proof (CUP): Motivation

Motivating Example

$$\forall xy.\ \mathsf{from}\ (s\ x)\ y \rightarrow \mathsf{from}\ x\ [x \mid y]$$

Approach: CUP

- We need a representation for the stream.
- Let **f** N denote the stream: N, s N, s (s N), s (s (s N)), $\cdots$
- Later we will give **f** as a (higher-order) λ-term.
- Then **f** (s N) denotes the stream: s N, s (s N), s (s (s N)), $\cdots$
- So we have **f** N $\equiv$ [N | **f** (s N)]
- where $\equiv$ denotes equality.

Coinductive Uniform Proof (CUP): Motivation

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

Approach: CUP

- . . . we have $\mathbf{f}\ \mathsf{N} \equiv [\ \mathsf{N} \mid \mathbf{f}\ (\mathsf{s}\ \mathsf{N})\ ]$
- Let Co-I be: $\forall x.\ \text{from}\ x\ (\mathbf{f}\ x)$
- CUP (sketch):

Step 1 $\forall x.\ \text{from}\ x\ (\mathbf{f}\ x) \rightarrowtail \text{from}\ c\ (\mathbf{f}\ c)$
Step 2 $\text{from}\ c\ (\mathbf{f}\ c) \equiv \text{from}\ c\ [c \mid \mathbf{f}\ (\mathsf{s}\ c)]$
Step 3 $\text{from}\ c\ [c \mid \mathbf{f}\ (\mathsf{s}\ c)] \rightsquigarrow \text{from}\ (\mathsf{s}\ c)\ (\mathbf{f}\ (\mathsf{s}\ c))$ ✓

- $\text{from}\ (\mathsf{s}\ c)\ (\mathbf{f}\ (\mathsf{s}\ c))$ is an instance of Co-I, with substitution $[s\ c/x]$.

Coinductive Uniform Proof (CUP): Motivation

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

Approach: CUP

- Using the CUP proof, we can recover the SLD-derivation for an arbitrary instance from t (**f** t) of the Co-I: $\forall x.$ from x (**f** x).
  1. The substitution is $[t/x]$ when we get the instance from the Co-I.
  - Recall the CUP proof: $\forall x.$ from x (**f** x) $\rightharpoonup$ from c (**f** c) $\equiv$ from c $[c \mid \mathbf{f}\ (\mathsf{s}\ c)] \rightsquigarrow$ from (s c) (**f** (s c))
  - We need the segment κ : from c (**f** c) — from (s c) (**f** (s c))
  2. The substitution is $[s\ c/x]$ when we apply Co-I to terminate the proof.
  3. Using substitutions $[t/x]$ and $[s\ c/x]$, we can generate an *infinite* set Θ of substitutions $[t/c, s\ t/c, s(s\ t)/c, s(s(s\ t))/c, \ldots]$
  4. We assemble all members of $\{\kappa\sigma \mid \sigma \in \Theta\}$ to get: from t (**f** t) — from (s t) (**f** (s t)) — from ($s(s\ t)$) (**f** ($s(s\ t)$)) $\cdots$
  5. ... which is just the SLD-derivation (replacing — by $\rightsquigarrow$)

Coinductive Uniform Proof (CUP): Motivation

Motivating Example

$$\forall xy.\ \text{from}\ (s\ x)\ y \rightarrow \text{from}\ x\ [x \mid y]$$

Approach: CUP

- The pattern of SLD-derivation is captured by the CUP proof.
- The Co-I ($\forall x.\ \text{from}\ x\ (\mathbf{f}\ x)$) is a more general answer than that (from x y where $[(s\ x)/x, [x \mid y]/y]$) given by CoLP.

Coinductive Uniform Proof: Overview

- To represent **f**, we need fixed-point terms
- To prove universally quantified Co-I, we need hereditary Harrop formula and uniform proof.
- To apply the Co-I in later stage of the proof, we need a coinductive proof principle
- To prevent unsound application of Co-I, we need a guarding mechanism
- The system is sound w.r.t
  1. The greatest fixed-point model
  2. Intuitionistic sequent calculus extended with later modality.

Overview of Term Syntax

The Set Λ_Σ of Well Formed Terms on Σ

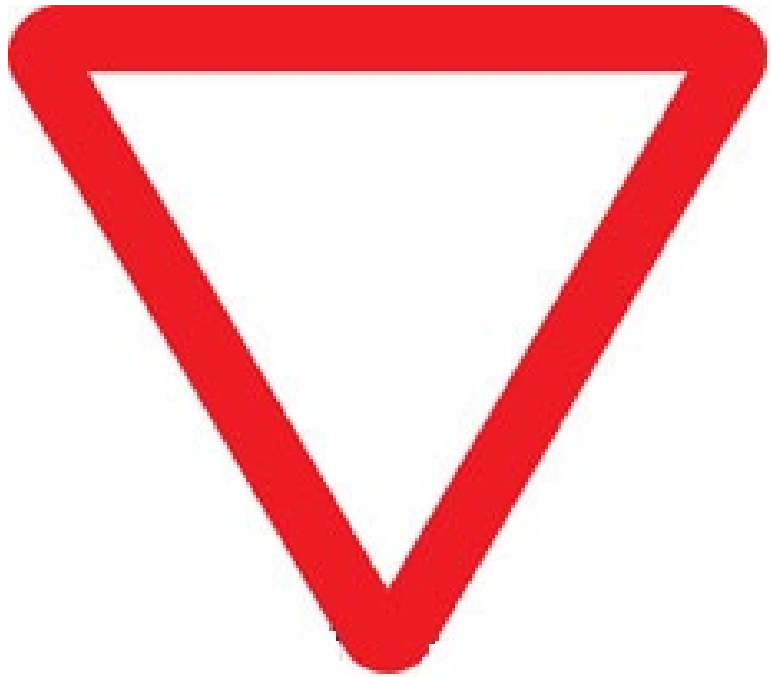
- Simply typed λ-terms extended with the fix binder to denote fixed-points.

$$\frac{\Sigma; \Gamma, x : \tau \vdash M : \tau}{\Sigma; \Gamma \vdash \mathrm{fix}\, x.\, M : \tau} \quad \text{compare with:} \quad \frac{\Sigma; \Gamma, x : \sigma \vdash M : \tau}{\Sigma; \Gamma \vdash \lambda x.\, M : \sigma \to \tau}$$

- $\mathrm{fix}\, x.\, M$ is supposed to be equal to $M\,[\mathrm{fix}\, x.\, M / x]$.

The Set Λ_Σ^G of Guarded Well Formed Terms

- Guarded terms are particular well formed terms.
- A guarded term models either a finite *or* an infinite term that occurs in first-order Horn clause logic programming.



Low level details ahead

The Type System

Definition

$\mathbb{B}$ — The set of *base type*. $o \notin \mathbb{B} = \{\iota\}$.

$\mathbb{T}$ — The set of *(simple) types*. $\tau \in \mathbb{T} ::= \mathbb{B} \mid \mathbb{B} \to \mathbb{T}$

$\mathbb{P}$ — The set of *proposition types*. $\rho \in \mathbb{P} ::= o \mid \mathbb{B} \to \mathbb{P}$

- We adopt the usual convention that $\to$ binds to the right.

Order $\mathsf{ord}(\iota) = \mathsf{ord}(o) = 0$; all other types $\pi \in \mathbb{T} \cup \mathbb{P}$ have $\mathsf{ord}(\pi) = 1$.

Arity $\mathsf{ar}(\iota) = \mathsf{ar}(o) = 0$; if $\pi \in \mathbb{T} \cup \mathbb{P}$ then $\mathsf{ar}(\iota \to \pi) = \mathsf{ar}(\pi) + 1$.

Example

$\mathbb{T} = \{\iota,\ \iota \to \iota,\ \iota \to \iota \to \iota,\ \ldots\}$. $\mathbb{P} = \{o,\ \iota \to o,\ \iota \to \iota \to o,\ \ldots\}$.
In other words, any $\tau \in \mathbb{T}$ can be depicted as $\iota^{\mathsf{ar}(\tau)} \to \iota$, any $\rho \in \mathbb{P}$ can be depicted as $\iota^{\mathsf{ar}(\rho)} \to o$.

Signature and Context

Definition

Con — A countable set of constants $a, b, c, \ldots$.

Var — A countable set of variables $x, y, z, \ldots$.

Σ — A *signature*, $\mathrm{Con} \mapsto (\mathbb{T} \cup \mathbb{P})$.

- $\Sigma_{\mathbb{T}}$ — The set of *term symbols* in Σ with types in $\mathbb{T}$.
  - $\Sigma^n_{\mathbb{T}}$ is the subset $\{c : \tau \in \Sigma_{\mathbb{T}} \mid \mathrm{ord}(\tau) \leq n\}$ of $\Sigma_{\mathbb{T}}$.
- $\Sigma_{\mathbb{P}}$ — The set of *predicate symbols* in Σ with types in $\mathbb{P}$.
  - $\Sigma^n_{\mathbb{P}}$ is the subset $\{r : \rho \in \Sigma_{\mathbb{P}} \mid \mathrm{ord}(\rho) \leq n\}$ of $\Sigma_{\mathbb{P}}$.

Γ — A *context*, $\mathrm{Var} \mapsto \mathbb{T}$.

- $\Gamma_{\mathbb{T}}$ — A synonym of Γ.
  - $\Gamma^n_{\mathbb{T}}$ is the subset $\{x : \tau \in \Gamma_{\mathbb{T}} \mid \mathrm{ord}(\tau) \leq n\}$ of $\Gamma_{\mathbb{T}}$.

Example

Let $\Sigma = \{a : \iota\}$ then $\Sigma_{\mathbb{T}} = \Sigma^1_{\mathbb{T}} = \Sigma^0_{\mathbb{T}} \ni a$

Let $\Gamma = \{y : \iota \to \iota\}$ then $\Gamma_{\mathbb{T}} = \Gamma^1_{\mathbb{T}} \ni y \notin \Gamma^0_{\mathbb{T}} = \varnothing$

The Set Λ_Σ of Well Formed Terms on Σ

Definition

$M \in \Lambda_\Sigma$ iff $\Sigma; \Gamma \vdash_{(m;n)} M : \tau$ for some order constraints $m, n \geq 0$, and $\tau \in \mathbb{T}$. We write $\Sigma; \Gamma \vdash^*_{(m;n)} M : \tau$ only if $\Sigma; \Gamma \vdash_{(m;n)} M : \tau$ and M does *not* contain any of $\{\mathrm{fix}, \lambda\}$.

$$\frac{c : \tau \in \Sigma^m_{\mathbb{T}}}{\Sigma; \Gamma \vdash_{(m;n)} c : \tau} \qquad \frac{x : \tau \in \Gamma^n_{\mathbb{T}}}{\Sigma; \Gamma \vdash_{(m;n)} x : \tau}$$

$$\frac{\Sigma; \Gamma \vdash_{(m;n)} M : \sigma \to \tau \qquad \Sigma; \Gamma \vdash_{(m;n)} N : \sigma}{\Sigma; \Gamma \vdash_{(m;n)} M\ N : \tau}$$

$$\frac{\Sigma; \Gamma, x : \sigma \vdash_{(m;n)} M : \tau}{\Sigma; \Gamma \vdash_{(m;n)} \lambda x.\, M : \sigma \to \tau} \qquad \frac{\Sigma; \Gamma, x : \tau \vdash_{(m;n)} M : \tau}{\Sigma; \Gamma \vdash_{(m;n)} \mathrm{fix}\, x.\, M : \tau}$$

Figure: Definition of $\Sigma; \Gamma \vdash_{(m;n)} M : \tau$.

The Set Λ_Σ of Well Formed Terms on Σ

Example

- Let $\Sigma = \{a : \iota, f : \iota \to \iota\}, \Gamma = \{y : \iota \to \iota\}$.
- Provable: $\Sigma; \Gamma \vdash_{(1;1)} y\ a : \iota$
- Not provable: $\Sigma; \Gamma \vdash_{(1;0)} y\ a : \iota$

↑ Mind the order constraints.

- Provable: $\begin{cases} \Sigma; \varnothing \vdash_{(1;0)} \lambda x.\, f\ x : \iota \to \iota \\ \Sigma; \varnothing \vdash_{(1;0)} \mathrm{fix}\, x.\, f\ x : \iota \end{cases}$
- Not provable: $\begin{cases} \Sigma; \varnothing \vdash^*_{(1;0)} \lambda x.\, f\ x : \iota \to \iota \\ \Sigma; \varnothing \vdash^*_{(1;0)} \mathrm{fix}\, x.\, f\ x : \iota \end{cases}$

↑ Mind the $*$, and note that $\lambda x.\, f\ x$ and $\mathrm{fix}\, x.\, f\ x$ contain the binders fix, λ.

The Set $\Lambda^{\mathcal{G}}_{\Sigma}$ of Guarded Well Formed Terms

Definition

If $\Sigma; \varnothing \vdash_{\triangleright} M : \tau$ then M is a *guarded fixed-point*. If $\Sigma; \Gamma \vdash_{g} M : \iota$, then M is a *guarded well formed term*. We denote the set of all guarded well formed terms on Σ by $\Lambda^{\mathcal{G}}_{\Sigma}$.

$$\frac{\Sigma; \Gamma \vdash^{*}_{(1;0)} M : \iota}{\Sigma; \Gamma \vdash_{g} M : \iota}$$

$$\frac{\Sigma; \varnothing \vdash_{\triangleright} M : \tau \qquad \mathsf{ar}(\tau) = |\vec{N}| \qquad \{\Sigma; \Gamma \vdash^{*}_{(1;0)} N : \iota \mid N \in \vec{N}\}}{\Sigma; \Gamma \vdash_{g} M\ \vec{N} : \iota}$$

$$\frac{\{\Sigma; \vec{x} : \iota \vdash^{*}_{(1;0)} N : \iota \mid N \in \vec{N}_{1(2,3)}\} \qquad \left[\begin{array}{ll} f : \tau' \in \Sigma^{1}_{\mathbb{T}} & \mathsf{ar}(\tau') = |\vec{N}_1| + 1 + |\vec{N}_3| \\ y : \tau \notin \vec{x} & \mathsf{ar}(\tau\ ) = |\vec{x}| = |\vec{N}_2| \end{array}\right]}{\Sigma; \varnothing \vdash_{\triangleright} \mathrm{fix}\, y.\, \lambda \vec{x}.\, f\ \vec{N}_1\ (y\ \vec{N}_2)\ \vec{N}_3 : \tau}$$

Figure: Definition of $\Sigma; \Gamma \vdash_{g} M : \tau$ and $\Sigma; \Gamma \vdash_{\triangleright} M : \tau$

The Set $\Lambda^{\mathcal{G}}_{\Sigma}$ of Guarded Well Formed Terms

Example

- Recall: we let **f** z denote the stream:
  z, s z, s (s z), s (s (s z)), $\cdots$
- Now we give **f** as $\mathrm{fix}\, y.\, \lambda x.\, [x \mid y\ (s\ x)]$.
- We justify this definition later using the notion of *reductions*.
- Let $\Sigma = \{[\_ \mid \_] : \iota \to \iota \to \iota, s : \iota \to \iota\}$, we have

$$\Sigma; \varnothing \vdash_{\triangleright} \mathbf{f} : \iota \to \iota$$

and

$$\Sigma; z : \iota \vdash_{g} \mathbf{f}\ z : \iota$$

The Set $\Lambda^{\mathcal{G}}_{\Sigma}$ of Guarded Well Formed Terms

Note that:

- By Def. of $\Lambda^{\mathcal{G}}_{\Sigma}$, there is *at most one variable* $y : \tau$ bound by fix within any given $M \in \Lambda^{\mathcal{G}}_{\Sigma}$.
- By Def. of $\mathbb{T} \ni \tau$, $\mathrm{ord}(\tau)$ can only be 0 or 1.

Definition

- $M \in \Lambda^{\mathcal{G}}_{\Sigma}$ is first-order if either 1) M does not contain fix, or 2) there exist $y : \tau$ fix-bound in M and $\mathrm{ord}(\tau) = 0$.
- $M \in \Lambda^{\mathcal{G}}_{\Sigma}$ is higher-order if there exist $y : \tau$ fix-bound in M and $\mathrm{ord}(\tau) = 1$.

Example

- $\mathbf{f}\, z$, i.e. $\mathrm{fix}\, y : \iota \to \iota.\, \lambda x.\, [x \mid y\ (s\ x)]\, z$ is higher-order.
- $\mathrm{fix}\, y : \iota.\, [0 \mid y]$ is first-order.

Well Formed Formulae

Definition

φ is a *atomic formula* on Σ if $\Sigma; \Gamma \Vdash_a \varphi$ for some Γ; φ is a *well formed formula* (wff) on Σ if $\Sigma; \Gamma \Vdash \varphi$ for some Γ. A wff φ is closed if $\Sigma; \varnothing \Vdash \varphi$.

$$\frac{(p : \iota^n \to o) \in \Sigma^1_{\mathbb{P}} \quad \{\Sigma; \Gamma \vdash_g M_k : \iota \mid 1 \leq k \leq n\}}{\Sigma; \Gamma \Vdash_a p\, M_1 \cdots M_n}$$

$$\frac{\Sigma; \Gamma \Vdash_a \varphi}{\Sigma; \Gamma \Vdash \varphi} \qquad \frac{\Gamma, x : \iota \Vdash \varphi}{\Sigma; \Gamma \Vdash \forall x : \iota. \varphi} \qquad \frac{\Gamma, x : \iota \Vdash \varphi}{\Sigma; \Gamma \Vdash \exists x : \iota. \varphi}$$

$$\frac{\Sigma; \Gamma \Vdash \varphi \qquad \Sigma; \Gamma \Vdash \psi \qquad \square \in \{\wedge, \vee, \to\}}{\Sigma; \Gamma \Vdash \varphi \square \psi}$$

Figure: Formulae

Well Formed Formulae

Definition

A well formed formula φ is first-order is all terms involved are first-order. Otherwise φ is higher-order.

Example

- $\forall \vec{x} : \iota.$ from $(s\ x_1)\ x_2 \rightarrow$ from $x_1\ [x_1 \mid x_2]$ is first-order (and closed).
- $\forall x : \iota.$ from $x\ (\mathbf{f}\, x)$, where $\mathbf{f}$ is $\mathrm{fix}\, y : \iota \rightarrow \iota.\, \lambda z.\, [z \mid y\ (s\ z)]$, is higher-order (and closed).

Hereditary Harrop Formula for Coinductive Uniform Proof

A — The set of atomic formulae on Σ.

G — The set of well formed hereditary Harrop goal formulae.

$$G ::= A \mid G \wedge G \mid G \vee G \mid \exists x : \iota.\, G \mid D \to G \mid \forall x : \iota.\, G$$

D — The set of well formed hereditary Harrop program formulae.

$$D ::= A \mid G \to D \mid D \wedge D \mid \forall x : \iota.\, D$$

(G', D') The pair of subsets of G and D containg all and only closed formulae.

- We take (G', D') as the abstract language for coinductive uniform proof.

Hereditary Harrop Formula for Coinductive Uniform Proof

Definition

- A program is a subset of D'.
- A goal is a member of G'.

Example

The two formulae below consist of a program:

1. $\forall \vec{x} : \iota.$ from $(s\ x_1)\ x_2 \rightarrow$ from $x_1\ [x_1 \mid x_2]$
2. $\forall x : \iota.$ from $x\ (\mathbf{f}\, x)$

Either formula above can be a goal.

Equivalence Relation for Terms and Formulae

Definition

On terms in Λ_Σ:

- *β-reduction* ($\longrightarrow_\beta$): $(\lambda x.\, M)N \longrightarrow_\beta M\,[N/x]$
- *fix-reduction* ($\longrightarrow_{\mathrm{fix}}$): $(\mathrm{fix}\, x.\, M) \longrightarrow_{\mathrm{fix}} M\,[\mathrm{fix}\, x.\, M/x]$
- *combined reduction* ($\longrightarrow$): The union of the compatible closures (reductions under applications and binders) of $\longrightarrow_\beta$ and $\longrightarrow_{\mathrm{fix}}$.
- *convertible relation* ($\equiv$): The equivalence closure of $\longrightarrow$.
- *convertible atoms*: Two atoms $p\, M_1 \cdots M_n \equiv p\, M'_1 \cdots\, M'_n$ if $M_k \equiv M'_k$ for $k = 1, \ldots, n$.

Equivalence Relation for Terms and Formulae

Example

We use $\mathbf{f}$ to abbreviate $\mathrm{fix}\, y.\, \lambda x.\, [x \mid y\ (s\ x)]$. The following terms are convertible ($\equiv$).

- $\mathbf{f}\, z$
- $[z \mid \mathbf{f}\, (s\ z)]$
- $[z, s\ z \mid \mathbf{f}\, (s\ (s\ z))]$
- $[z, s\ z, s\ (s\ z) \mid \mathbf{f}\, (s\ (s\ (s\ z)))]$
- $\ldots$

This justifies our representation of the stream
z, s z, s (s z), s (s (s z)), $\cdots$ by $\mathbf{f}\, z$.

Coinductive Proof Principle

- $\Sigma; P; \Delta \Longrightarrow \varphi$ means φ has a uniform proof w.r.t program $P \cup \Delta$ on Σ.
- $\Sigma; P \Rightarrowtail \varphi$ means φ is coinductively provable from program P on Σ. φ is called a coinductive invariant.
- The rule for $\Sigma; P \Rightarrowtail \varphi$ is:

$$\frac{\Sigma; P; \varphi \Longrightarrow \langle \varphi \rangle}{\Sigma; P \Rightarrowtail \varphi}\ \textsc{co-fix}$$

where $\varphi \in M'$, $\langle\ \rangle$ regulates the proof of $\Sigma; P; \varphi \Longrightarrow \varphi$.

Reads If $\Sigma; P; \varphi \Longrightarrow \varphi$ in a regulated way, then $\Sigma; P \Rightarrowtail \varphi$.

M — The intersection of D and G, given as

$$M ::= A \mid M \wedge M \mid M \to M \mid \forall x : \iota.\, M$$

M' is the subset of M containing all and only closed formulae.

Uniform Proof

- Developed by Dale Miller et al in 1990s
- The top-level logical constant in a goal determines the goal(s) to prove next.
- A proof theoretic foundation for logic programming
- A criterion to judge logic programming languages.
  - A language L is suitable for logic programming, if the proposition below is true.
  - There is a uniform proof in L iff there is an intuitionistic proof in L.
- Four languages satisfy this criterion: first/higher-order Horn clause/hereditary Harrop formula
- No fixed-point terms. More complex type system.

Uniform Proof

$$\frac{\Sigma; P; \Delta \overset{D}{\Longrightarrow} A \qquad D \in P \cup \Delta}{\Sigma; P; \Delta \Longrightarrow A}\ \text{DECIDE} \qquad \frac{A \equiv A'}{\Sigma; P; \Delta \overset{A'}{\Longrightarrow} A}\ \text{INITIAL}$$

$$\frac{\Sigma; P; \Delta \overset{D}{\Longrightarrow} A \qquad \Sigma; P; \Delta \Longrightarrow G}{\Sigma; P; \Delta \overset{G \to D}{\Longrightarrow} A} \to L \qquad \frac{\Sigma; P, D; \Delta \Longrightarrow G}{\Sigma; P; \Delta \Longrightarrow D \to G} \to R$$

$$\frac{\Sigma; P; \Delta \overset{D_x}{\Longrightarrow} A \qquad x \in \{1, 2\}}{\Sigma; P; \Delta \overset{D_1 \wedge D_2}{\Longrightarrow} A} \wedge L \qquad \frac{\Sigma; P; \Delta \Longrightarrow G_1 \qquad \Sigma; P; \Delta \Longrightarrow G_2}{\Sigma; P; \Delta \Longrightarrow G_1 \wedge G_2} \wedge R$$

$$\frac{\Sigma; P; \Delta \overset{D[N/x]}{\Longrightarrow} A \qquad \Sigma; \varnothing \vdash_g N : \iota}{\Sigma; P; \Delta \overset{\forall x:\iota.\, D}{\Longrightarrow} A} \forall L \qquad \frac{c : \iota, \Sigma; P; \Delta \Longrightarrow G\,[c/x] \qquad c : \iota \notin \Sigma}{\Sigma; P; \Delta \Longrightarrow \forall x : \iota.\, G} \forall R$$

$$\frac{\Sigma; P; \Delta \Longrightarrow G\,[N/x] \qquad \Sigma; \varnothing \vdash_g N : \iota}{\Sigma; P; \Delta \Longrightarrow \exists x : \iota.\, G} \exists R \qquad \frac{\Sigma; P; \Delta \overset{G_x}{\Longrightarrow} A \qquad x \in \{1, 2\}}{\Sigma; P; \Delta \Longrightarrow G_1 \vee G_2} \vee R$$

Figure: Uniform Proof, with the field Δ for a coinductive invariant, and the relation $\equiv$ for equality between guarded terms.

Guarding Mechanism

$$\frac{\Sigma; P; \Delta \overset{D}{\Longrightarrow} A \qquad P \ni D \notin \Delta}{\Sigma; P; \Delta \Longrightarrow \langle A \rangle}\ \text{DECIDE}\langle\rangle$$

$$\frac{c : \iota, \Sigma; P; \Delta \Longrightarrow \langle M\,[\,c/x\,]\rangle \qquad c : \iota \notin \Sigma}{\Sigma; P; \Delta \Longrightarrow \langle \forall x : \iota.\ M\rangle}\ \forall R\langle\rangle$$

$$\frac{\Sigma; P; \Delta \Longrightarrow \langle M_1\rangle \qquad \Sigma; P; \Delta \Longrightarrow \langle M_2\rangle}{\Sigma; P; \Delta \Longrightarrow \langle M_1 \wedge M_2\rangle}\ \wedge R\langle\rangle \qquad \frac{\Sigma; P, M_1; \Delta \Longrightarrow \langle M_2\rangle}{\Sigma; P; \Delta \Longrightarrow \langle M_1 \to M_2\rangle}\ \to R\langle\rangle$$

Figure: Guarding Mechanism

Soundness Properties: w.r.t Herbrand Model

CUP is sound w.r.t the greatest fixed-point model M_ν.

Theorem

If $\Sigma; P \looparrowright \varphi$ *then* $M_\nu \models \varphi$.

Proof Sketch.

- A coinductive uniform proof is a template.
- Using certain substitutions involved in the proof,
- an infinite amount of substitutions can be generated,
- which can instantiate the template into an infinite amount of instances
- The infinite SLD-derivation can be obtained by assembling these instances.

□

Soundness Properties: w.r.t Herbrand Model

CUP is sound w.r.t the greatest fixed-point model M_ν.

Theorem

If $\Sigma; P \upharpoonright\!\!\rightarrow \varphi$ *and* $\Sigma; P, \varphi \upharpoonright\!\!\rightarrow \psi$, *then* $M_\nu \models \psi$
— *provided* φ *either has no* $\forall$ *or has no* $\rightarrow$.

Proof Sketch.

Since $\Sigma; P \upharpoonright\!\!\rightarrow \varphi$, we have $M_\nu \models \varphi$. Let M'_ν be the greatest fixed-point model of $P \cup \{\varphi\}$. Since $\Sigma; P, \varphi \upharpoonright\!\!\rightarrow \psi$, we have $M'_\nu \models \psi$. We show that $M_\nu = M'_\nu$. □

- If φ involves both $\rightarrow$ and $\forall$, we may still use φ as a lemma, provided some further conditions are satisfied.

Soundness Properties: w.r.t iFOL$_\blacktriangleright$

CUP is sound w.r.t intuitionistic sequent calculus extended with later modality (**iFOL$_\blacktriangleright$**)

Definition

The formulae of the logic **iFOL$_\blacktriangleright$** over Σ are well formed formulae extended with the following rule. Conversion ($\equiv$) extends to these formulae in the obvious way.

$$\frac{\Sigma; \Gamma \Vdash \varphi}{\Sigma; \Gamma \Vdash \blacktriangleright \varphi}$$

Definition

$\Gamma \mid \Delta \vdash \varphi$ means the formula φ is provable in context Γ w.r.t the set Δ of formulae.

Soundness Properties: w.r.t iFOL$_\blacktriangleright$

$$\frac{\Sigma; \Gamma \Vdash \Delta \qquad \varphi \in \Delta}{\Gamma \mid \Delta \vdash \varphi}\ \textbf{(Proj)} \qquad \frac{\Gamma \mid \Delta \vdash \varphi' \qquad \varphi \equiv \varphi'}{\Gamma \mid \Delta \vdash \varphi}\ \textbf{(Conv)}$$

$$\frac{\Gamma \mid \Delta \vdash \varphi \qquad \Gamma \mid \Delta \vdash \psi}{\Gamma \mid \Delta \vdash \varphi \wedge \psi}\ (\wedge\textbf{-I}) \qquad \frac{\Gamma \mid \Delta \vdash \varphi_1 \wedge \varphi_2 \qquad i \in \{1,2\}}{\Gamma \mid \Delta \vdash \varphi_i}\ (\wedge_i\textbf{-E})$$

$$\frac{\Gamma \mid \Delta \vdash \varphi_i \qquad i \in \{1,2\}}{\Gamma \mid \Delta \vdash \varphi_1 \vee \varphi_2}\ (\vee_i\textbf{-I}) \qquad \frac{\Gamma \mid \Delta, \varphi_1 \vdash \psi \qquad \Gamma \mid \Delta, \varphi_2 \vdash \psi}{\Gamma \mid \Delta, \varphi_1 \vee \varphi_2 \vdash \psi}\ (\vee\textbf{-E})$$

$$\frac{\Gamma \mid \Delta, \varphi \vdash \psi}{\Gamma \mid \Delta \vdash \varphi \to \psi}\ (\to\textbf{-I}) \qquad \frac{\Gamma \mid \Delta \vdash \varphi \to \psi \qquad \Gamma \mid \Delta \vdash \varphi}{\Gamma \mid \Delta \vdash \psi}\ (\to\textbf{-E})$$

$$\frac{\Gamma, x : \tau \mid \Delta \vdash \varphi \qquad x : \tau \notin \Gamma}{\Gamma \mid \Delta \vdash \forall x : \tau.\, \varphi}\ (\forall\textbf{-I}) \qquad \frac{\Gamma \mid \Delta \vdash \forall x : \tau.\, \varphi \qquad \Sigma; \Gamma \vdash_{(m;n)} M : \tau}{\Gamma \mid \Delta \vdash \varphi\,[M/x]}\ (\forall\textbf{-E})$$

$$\frac{\Sigma; \Gamma \vdash_{(m;n)} M : \tau \qquad \Gamma \mid \Delta \vdash \varphi\,[M/x]}{\Gamma \mid \Delta \vdash \exists x : \tau.\, \varphi}\ (\exists\textbf{-I}) \qquad \frac{\Gamma, x : \tau \mid \Delta, \varphi \vdash \psi \qquad x : \tau \notin \Gamma}{\Gamma \mid \Delta, \exists x : \tau.\, \varphi \vdash \psi}\ (\exists\textbf{-E})$$

Intuitionistic Rules for Standard Connectives

$$\frac{\Gamma \mid \Delta \vdash \varphi}{\Gamma \mid \Delta \vdash \blacktriangleright \varphi}\ \textbf{(Next)} \qquad \frac{\Gamma \mid \Delta \vdash \blacktriangleright(\varphi \to \psi)}{\Gamma \mid \Delta \vdash \blacktriangleright \varphi \to \blacktriangleright \psi}\ \textbf{(Mon)} \qquad \frac{\Gamma \mid \Delta, \blacktriangleright \varphi \vdash \varphi}{\Gamma \mid \Delta \vdash \varphi}\ \textbf{(Löb)}$$

Rules for the Later Modality

Soundness Properties: w.r.t iFOL$_\blacktriangleright$

Definition

Given a Horn clause φ of the shape $\forall \vec{x}. (A_1 \wedge \cdots \wedge A_n) \to A$, we define its *guarding* $\overline{\varphi}$ to be $\forall \vec{x}. (\blacktriangleright A_1 \wedge \cdots \wedge \blacktriangleright A_n) \to A$. For a collection P of Horn clauses, we define its guarding $\overline{P}$ by guarding each formula in P.

Theorem

If $\Sigma; P \not\hookrightarrow \varphi$ *then* $\varnothing \mid \overline{P} \vdash \varphi$.

Proof Sketch.

We do case analysis with an inductive argument. □

Summary

Acknowledgment

We thank Dr Murdoch James Gabbay for his suggestions to improve these slides !


UNIVERSITÉ
DE LYON


ENS DE LYON


HERIOT
WATT
UNIVERSITY