Kleene realizability and negative translations

Alexandre Miquel

UNIVERSIDAD DE LA REPUBLICA URUGUAY

FACULTAD DE INGENIERIA

EQUIPO DE LÓGICA UDELAR

October 27th, 2021

Plan

1. Kleene realizability
2. Gödel-Gentzen negative translation
3. Uniformity and relativization
4. Lafont-Reus-Streicher negative translation

Plan

1. Kleene realizability
2. Gödel-Gentzen negative translation
3. Uniformity and relativization
4. Lafont-Reus-Streicher negative translation

The language of realizers (recall)

Terms of system T (= λ-calculus + primitive pairs & integers)

$$\begin{array}{llcl} \lambda\textbf{-terms} & t, u & ::= & x \mid \lambda x\,.\,t \mid tu \\ & & \mid & \langle t_1, t_2\rangle \mid \pi_1(t) \mid \pi_2(t) \\ & & \mid & 0 \mid \mathtt{S}(t) \mid \mathtt{rec}(t_0, t_1, u) \end{array}$$

Syntactic worship: Free & bound variables. Renaming. Work up to α-conversion. Set of free variables: $FV(t)$. Capture-avoiding substitution: $t[x := u]$

- **Notation:** $\bar{n} \;:\equiv\; \mathtt{S}^n\,0 \quad (n \in \mathbb{N})$

Reduction rules

$$(\lambda x\,.\,t)\,u \;\succ\; t[x := u]$$

$$\begin{array}{lcl} \pi_1(\langle t_1, t_2\rangle) & \succ & t_1 \\ \pi_2(\langle t_1, t_2\rangle) & \succ & t_2 \end{array} \qquad \begin{array}{rcl} \mathtt{rec}(t_0,\, t_1,\, 0) & \succ & t_0 \\ \mathtt{rec}(t_0,\, t_1\, \mathtt{S}(u)) & \succ & t_1\;\, u\; (\mathtt{rec}(t_0, t_1, u)) \end{array}$$

- Grand reduction written $t \succ^* u$ (reflexive, transitive, context-closed)

Definition of the relation $t \Vdash A$ (recall)

- **Recall:** For each closed FO-term e, we write $e^{\mathbb{N}}$ its denotation in $\mathbb{N}$

Definition of the realizability relation $t \Vdash A$ (t, A closed)

$$
\begin{array}{lcl}
t \Vdash \bot & \equiv & \bot \\
t \Vdash \top & \equiv & t \succ^* 0 \\
t \Vdash e_1 = e_2 & \equiv & e_1^{\mathbb{N}} = e_2^{\mathbb{N}} \ \wedge \ t \succ^* 0 \\
t \Vdash A \wedge B & \equiv & \exists t_1\, \exists t_2\, (t \succ^* \langle t_1, t_2 \rangle \ \wedge \ t_1 \Vdash A \ \wedge \ t_2 \Vdash B) \\
t \Vdash A \vee B & \equiv & \exists u\, ((t \succ^* \langle \bar{0}, u \rangle \ \wedge \ u \Vdash A) \ \vee \ (t \succ^* \langle \bar{1}, u \rangle \ \wedge \ u \Vdash B)) \\
t \Vdash A \Rightarrow B & \equiv & \forall u\, (u \Vdash A \ \Rightarrow \ tu \Vdash B) \\
t \Vdash \forall x\, A(x) & \equiv & \forall n\, (t\, \bar{n} \Vdash A(n)) \\
t \Vdash \exists x\, A(x) & \equiv & \exists n\, \exists u\, (t \succ^* \langle \bar{n}, u \rangle \ \wedge \ u \Vdash A(n))
\end{array}
$$

Lemma (Closure under anti-reduction)

If $t \succ^* t'$ and $t' \Vdash A$, then $t \Vdash A$

The main Theorem (recall)

Lemma (Adequacy)

Let $d : (A_1, \ldots, A_n \vdash B)$ be a derivation in NJ. Then:

- for all valuations ρ : FOVar $\to \mathbb{N}$,
- for all realizers $t_1 \Vdash A_1[\rho], \ldots, t_n \Vdash A_n[\rho]$,

we have: $d^*[\rho][z_1 := t_1, \ldots, z_n := t_n] \Vdash B[\rho]$

writing d^* the λ-term extracted from the derivation d (following Curry-Howard)

Lemma

All axioms of HA are realized

Theorem (Soundness)

If HA $\vdash A$, then $t \Vdash A$ for some closed λ-term t

Harrop formulas (1/2)

The class of Harrop formulas

Harrop formulas $H ::= e_1 = e_2 \mid \top \mid \bot \mid H_1 \wedge H_2 \mid A \Rightarrow H \mid \forall x\, H$

- **Intuition:** Harrop formulas do not contain the two "problematic" constructions $\vee$ and $\exists$, except on the left-hand side of implications
- Therefore, Harrop formulas are classical:

Proposition

For each Harrop formula $H(\vec{x})$:

$$\mathrm{HA} \vdash \forall \vec{x}\,(H(\vec{x}) \Leftrightarrow \neg\neg H(\vec{x}))$$

Proof. By structural induction on $H(\vec{x})$.

Harrop formulas (2/2)

- To each (possibly open) Harrop formula H, we associate a closed λ-term t_H that is computationally trivial:

$$\tau_H :\equiv 0 \quad (H \text{ atomic}) \qquad \tau_{A\Rightarrow H} :\equiv \lambda\_ .\, \tau_H$$
$$\tau_{H_1 \wedge H_2} :\equiv \langle \tau_{H_1}, \tau_{H_2} \rangle \qquad \tau_{\forall x\, H} :\equiv \lambda\_ .\, \tau_H$$

Theorem

For all closed Harrop formulas H:

$$\text{If} \quad H \text{ is realized}, \quad \text{then} \quad \tau_H \Vdash H$$

Moreover, all realizers of H are "computationally equivalent" to τ_H

- **Intuition:** Harrop formulas have computationally irrelevant realizers, that can be replaced by the trivial realizers τ_H
  - Useful for optimizing extracted programs (cf next slide)
  - But shows that Harrop formulas are computationally irrelevant

Optimizing program extraction (1/2)

Idea: While turning derivations intro λ-terms, use Harrop realizers τ_H whenever possible (instead of following Curry-Howard)

$\Rightarrow$ Optimized program extraction

Definition (Optimized program extraction)

Each derivation $d : (\Gamma \vdash B)$ is turned into a λ-term d^{opt} as follows:

- If B is a Harrop formula, then $d^{\text{opt}} \;:\equiv\; \tau_B$
- Otherwise, follow Curry-Howard for the last rule:
  - If $d \equiv \left\{ \dfrac{\begin{matrix} \vdots\ d_1 \\ \Gamma, A \vdash C \end{matrix}}{\Gamma \vdash A \Rightarrow C} \right.$ then $d^{\text{opt}} \;:\equiv\; \lambda z_A \,.\, d_1^{\text{opt}}$
  - If $d \equiv \left\{ \dfrac{\begin{matrix} \vdots\ d_1 \\ \Gamma \vdash A \Rightarrow B \end{matrix} \quad \begin{matrix} \vdots\ d_2 \\ \Gamma \vdash A \end{matrix}}{\Gamma \vdash B} \right.$ then $d^{\text{opt}} \;:\equiv\; d_1^{\text{opt}} d_2^{\text{opt}}$
  - etc.

Optimizing program extraction (2/2)

Lemma (Adequacy of the optimized extraction)

Let $d : (A_1, \ldots, A_n \vdash B)$ be a derivation in NJ. Then for all valuations $\rho : \mathrm{FOVar} \to \mathbb{N}$ and for all realizers $t_1 \Vdash A_1[\rho], \ldots, t_n \Vdash A_n[\rho]$, we have:

$$d^{\mathrm{opt}}[\rho][z_1 := t_1, \ldots, z_n := t_n] \Vdash B[\rho]$$

Example:

- Let $F :\equiv \forall x \forall y \forall z \forall n (n > 2 \Rightarrow x^n + y^n \neq z^n)$

  (Fermat's last theorem, as a Harrop formula)

- Given a derivation $d \equiv \left\{ \dfrac{\begin{array}{c} \vdots\ d_I \\ \vdash F \Rightarrow A \end{array} \quad \begin{array}{c} \vdots\ d_F \\ \vdash F \end{array}}{\vdash A} \right.$ (where A is not Harrop)

  we have: $d^{\mathrm{opt}} \equiv {d_I}^{\mathrm{opt}} d_F^{\mathrm{opt}} \equiv {d_I}^{\mathrm{opt}} \tau_F$

  $\equiv {d_I}^{\mathrm{opt}}(\lambda \_, \_, \_, \_, \_, \_ . 0) \Vdash A$

$\Rightarrow$ Don't need to know the proof of Fermat's last theorem to realize A!

Plan

1. Kleene realizability
2. Gödel-Gentzen negative translation
3. Uniformity and relativization
4. Lafont-Reus-Streicher negative translation

How to cope with classical logic?

- Kleene realizability is definitely incompatible with classical logic:

Proposition (cf previous talk)

$$\not\Vdash \quad \forall x\,(\mathsf{Halt}(x) \vee \neg\mathsf{Halt}(x))$$

$$\text{any\_term} \quad \Vdash \quad \neg\forall x\,(\mathsf{Halt}(x) \vee \neg\mathsf{Halt}(x))$$

(The same holds for all variants of Kleene realizability)

- Two possible solutions:

1. Compose Kleene realizability with a negative translation from classical logic (LK) to intuitionistic logic (LJ) (next slide)
2. Reformulate the principles of realizability to make them compatible with classical logic: Krivine classical realizability (next talk)

The Gödel-Gentzen negative translation

- **Idea:** Turn positive constructions (atomic formulas, $\vee$, $\exists$) into negative constructions ($\bot$, $\neg$, $\Rightarrow$, $\wedge$, $\forall$) using De Morgan laws
- Every formula A is translated into a formula $A^{\mathbf{G}}$ defined by:

$$
\begin{array}{rcl@{\qquad}rcl}
\top^{\mathbf{G}} & :\equiv & \top & \bot^{\mathbf{G}} & :\equiv & \bot \\
(A \Rightarrow B)^{\mathbf{G}} & :\equiv & A^{\mathbf{G}} \Rightarrow B^{\mathbf{G}} & (e_1 = e_2)^{\mathbf{G}} & :\equiv & \neg\neg(e_1 = e_2) \\
(A \wedge B)^{\mathbf{G}} & :\equiv & A^{\mathbf{G}} \wedge B^{\mathbf{G}} & (A \vee B)^{\mathbf{G}} & :\equiv & \neg(\neg A^{\mathbf{G}} \wedge \neg B^{\mathbf{G}}) \\
(\forall x\, A)^{\mathbf{G}} & :\equiv & \forall x\, A^{\mathbf{G}} & (\exists x\, A)^{\mathbf{G}} & :\equiv & \neg\forall x\, \neg A^{\mathbf{G}}
\end{array}
$$

writing: $\neg A \;:\equiv\; A \Rightarrow \bot$

Theorem (Soundness)

1. $\mathrm{LK} \vdash A^{\mathbf{G}} \Leftrightarrow A$
2. If $\mathrm{PA} \vdash A$, then $\mathrm{HA} \vdash A^{\mathbf{G}}$

Realizing translated formulas

- **Strategy:**
  1. Build a derivation d of A (in PA)
  2. Turn it into a derivation $d^{\mathbf{G}}$ of $A^{\mathbf{G}}$ (in HA)
  3. Turn $d^{\mathbf{G}}$ into a Kleene realizer (program extraction)
- Does not work! Failure comes from:

Proposition (Realizability collapse)

For every closed formula A:

1. $A^{\mathbf{G}}$ is a Harrop formula (computationally irrelevant)
2. Kleene's semantics for $A^{\mathbf{G}}$ mimics Tarski's semantics for A:

$$A^{\mathbf{G}} \text{ is realized} \quad \text{iff} \quad \tau_{A^{\mathbf{G}}} \Vdash A^{\mathbf{G}} \quad \text{iff} \quad \mathbb{N} \models A$$

Proof. By structural induction on A.

- **Conclusion:** Kleene $\circ$ Gödel-Gentzen $=$ Tarski (in $\mathbb{N}$)

Friedman's R-translation (called A-translation by Friedman)

- **Principle:** In Gödel-Gentzen translation, replace each occurrence of $\bot$ (absurdity) by a fixed formula R, called the return formula
  Note: The return formula R may contain free variables!

- Every formula A is translated into a formula $A^{\mathbf{F}}$ defined by:

$$
\begin{array}{rcl}
\top^{\mathbf{F}} & :\equiv & \top \\
(A \Rightarrow B)^{\mathbf{F}} & :\equiv & A^{\mathbf{F}} \Rightarrow B^{\mathbf{F}} \\
(A \wedge B)^{\mathbf{F}} & :\equiv & A^{\mathbf{F}} \wedge B^{\mathbf{F}} \\
(\forall x\, A)^{\mathbf{F}} & :\equiv & \forall x\, A^{\mathbf{F}} \\
 & & (\text{if } x \notin FV(R))
\end{array}
\qquad
\begin{array}{rcl}
\bot^{\mathbf{F}} & :\equiv & R \\
(e_1 = e_2)^{\mathbf{F}} & :\equiv & \neg_R \neg_R (e_1 = e_2) \\
(A \vee B)^{\mathbf{F}} & :\equiv & \neg_R(\neg_R A^{\mathbf{F}} \wedge \neg_R B^{\mathbf{F}}) \\
(\exists x\, A)^{\mathbf{F}} & :\equiv & \neg_R \forall x\, \neg_R A^{\mathbf{F}} \\
 & & (\text{if } x \notin FV(R))
\end{array}
$$

writing: $\neg_R A \;:\equiv\; A \Rightarrow R$

Theorem (Soundness)

If $\mathrm{PA} \vdash A$, then $\mathrm{HA} \vdash A^{\mathbf{F}}$ (independently from the formula R)

Beware! The formulas A and $A^{\mathbf{F}}$ are no more classically equivalent (in general)

Π^0_2-conservativity (1/2)

The interest of Friedman's translation comes from the following:

Theorem (Π^0_2-conservativity)

PA is a Π^0_2-conservative extension of HA, that is:

$$\mathrm{PA} \vdash \forall \vec{x}\, \exists \vec{y}\, f(\vec{x},\vec{y}) = 0 \qquad \text{iff} \qquad \mathrm{HA} \vdash \forall \vec{x}\, \exists \vec{y}\, f(\vec{x},\vec{y}) = 0$$

for every primitive recursive function $f(\vec{x},\vec{y})$

This more generally implies that:

$$\mathrm{PA} \vdash \forall \vec{x}\, \exists \vec{y}\, A(\vec{x},\vec{y}) \qquad \text{iff} \qquad \mathrm{HA} \vdash \forall \vec{x}\, \exists \vec{y}\, A(\vec{x},\vec{y})$$

for every formula $A(\vec{x},\vec{y})$ with bounded quantifications

Π^0_2-conservativity (2/2)

Proof. Assume that PA $\vdash \forall x\, \exists y\, f(x, y) = 0$.

Working with an unknown formula R, we observe that:

$$\begin{array}{lll} \text{HA} \vdash \forall x \neg_R \forall y \neg_R \neg_R \neg_R f(x,y) = 0 & & \text{(by } R\text{-translation)} \\ \text{HA} \vdash \forall x \neg_R \forall y \neg_R f(x,y) = 0 & & \text{(since } \neg_R\neg_R\neg_R \Leftrightarrow_{\text{LJ}} \neg_R) \\ \text{HA} \vdash \neg_R \forall y \neg_R f(x_0,y) = 0 & & \text{(by } \forall\text{-elim, with } x_0 \text{ fresh)} \\ \text{HA} \vdash \forall y\, (f(x_0,y) = 0 \Rightarrow R) \Rightarrow R & & \text{(from the def. of } \neg_R) \end{array}$$

We now take: $R :\equiv \exists y_0\, f(x_0, y_0) = 0$ (Friedman's trick!)

From the def. of R, we have:

$$\text{HA} \vdash \forall y\, (f(x_0,y) = 0 \Rightarrow \exists y_0\, f(x_0,y_0) = 0) \Rightarrow \exists y_0\, f(x_0,y_0) = 0$$

But the premise of the above implication is provable

$$\text{HA} \vdash \forall y\, (f(x_0,y) = 0 \Rightarrow \exists y_0\, f(x_0,y_0) = 0) \qquad \text{(by } \exists\text{-intro with } y_0 = y)$$

hence we get

$$\begin{array}{ll} \text{HA} \vdash \exists y_0\, f(x_0,y_0) = 0 & \text{(by modus ponens)} \\ \text{HA} \vdash \forall x_0\, \exists y_0\, f(x_0,y_0) = 0 & \text{(by } \forall\text{-intro)} \end{array}$$

The converse implication (HA $\vdash \cdots$ implies PA $\vdash \cdots$) is obvious. □

Realizing translated formulas, again

- **Strategy:**
  1. Build a derivation d of a Π^0_2-formula A (in PA)
  2. Turn it into a derivation F-trick(d^{F}) of A (in HA)
  3. Turn F-trick(d^{F}) into a Kleene realizer of A (program extraction)

- This technique perfectly works in practice. However:
  - The formula A^{F} is not a Harrop formula (in general), even when A is.
    Possible fix: Introduce specific optimization techniques, e.g.:

    Refined Program Extraction [Berger et al. 2001]

  - The translation $A \mapsto A^{\mathsf{F}}$ completely changes the structure of the underlying proof. **Possible fix:** cf next parts

Plan

1. Kleene realizability
2. Gödel-Gentzen negative translation
3. Uniformity and relativization
4. Lafont-Reus-Streicher negative translation

Uniform vs non-uniform quantifiers (1/2)

- In the Curry-Howard correspondence (and in realizability), there are two different ways to interpret quantifiers:

	$\forall x\, A(x)$	$\exists x\, A(x)$
Non-uniform (Type Theory style) (Kleene realiz.)	$\prod_{x\in D} A(x)$ (type of dep. functions)	$\sum_{x\in D} A(x)$ (type of dep. pairs)
Uniform (ML/Haskell style)	$\bigcap_{x\in D} A(x)$ (intersection type)	$\bigcup_{x\in D} A(x)$ (union type)

- **Remark:** Tarski/Kripke/Heyting/Cohen models do not distinguish the two interpretations: the difference only appears in realizability

Uniform vs non-uniform quantifiers (2/2)

- 1st-, 2nd- and higher-order logic support both interpretations
  (But uniform interpretation is more concise & natural)
- The same holds for impredicative set theories: ZF, IZF_C, IZF_R
- Arithmetic (PA/HA) only supports the non-uniform interpretation
  (due to the induction principle)
- But in all cases, the non-uniform interpretation can be encoded from the uniform interpretation, using a relativization:

$$\text{(non-uniform) } \forall x\, A(x) \quad :\equiv \quad \text{(uniform) } \forall x\, \underbrace{(D(x) \Rightarrow A(x))}_{\text{type of functions}}$$

$$\text{(non-uniform) } \exists x\, A(x) \quad :\equiv \quad \text{(uniform) } \exists x\, \underbrace{(D(x) \wedge A(x))}_{\text{type of pairs}}$$

where $D(x)$ is a suitable relativization predicate (the domain of quantification)

- This is why we shall prefer the uniform interpretation (in what follows)

Uniformity in realizability

- Kleene realizability interprets quantifiers in a non-uniform way:

$$
\begin{aligned}
t \Vdash \forall x\, A(x) &\quad\equiv\quad \forall n\, (t\, \bar{n} \Vdash A(n)) \\
t \Vdash \exists x\, A(x) &\quad\equiv\quad \exists n\, \exists u\, (t \succ^* \langle \bar{n}, u\rangle \;\wedge\; u \Vdash A(n))
\end{aligned}
$$

  - Realizers of $\forall x\, A(x)$ expect an argument (representing x)
  - Realizers of $\exists x\, A(x)$ bear a witness

- But realizability can also interpret quantifiers in a uniform way:

Definition of the uniform realizability relation $t \Vdash_{\mathrm{u}} A$

$$
\begin{aligned}
t \Vdash_{\mathrm{u}} \forall x\, A(x) &\quad:\equiv\quad \forall n\, (t \Vdash_{\mathrm{u}} A(n)) \\
t \Vdash_{\mathrm{u}} \exists x\, A(x) &\quad:\equiv\quad \exists n\, (t \Vdash_{\mathrm{u}} A(n))
\end{aligned}
$$

(other clauses of the definition are the same as for $\Vdash$)

  - Realizers of $\forall x\, A(x)$ do not expect an argument
  - Realizers of $\exists x\, A(x)$ do not bear a witness

- What does it change... in NJ? ... in HA?

The uniform interpretation of first-order logic (1/3)

Recall:

- To prove the adequacy of the rules of NJ w.r.t. the relation $t \Vdash A$ (where $\forall/\exists$ are interpreted non uniformly), we defined a translation

$$d : (A_1, \ldots, A_n \vdash B) \quad \mapsto \quad d^*$$

  where the λ-term d^* depends on the proof variables $z_{A_1}, \ldots, z_{A_n}$ and on the free variables $x_1, \ldots, x_k$ of the sequent $A_1, \ldots, A_n \vdash B$

- The rules for quantifiers were translated as follows:

$$\left(\frac{\begin{array}{c}\vdots\ d\\ \Gamma \vdash A\end{array}}{\Gamma \vdash \forall x\, A}\right)^* := \lambda x\,.\,d^* \qquad \left(\frac{\begin{array}{c}\vdots\ d\\ \Gamma \vdash \forall x\, A\end{array}}{\Gamma \vdash A[x := e]}\right)^* := d^*\,e^*$$

$$\left(\frac{\begin{array}{c}\vdots\ d\\ \Gamma \vdash A[x := e]\end{array}}{\Gamma \vdash \exists x\, A}\right)^* := \langle e^*, d^*\rangle \qquad \left(\frac{\begin{array}{cc}\vdots\ d_1 & \vdots\ d_2\\ \Gamma \vdash \exists x\, A & \Gamma, A \vdash B\end{array}}{\Gamma \vdash B}\right)^* := \text{let } \langle x, z\rangle = d_1^* \text{ in } d_2^*$$

The uniform interpretation of first-order logic (2/3)

- To prove the adequacy of the rules of NJ w.r.t. the relation $t \Vdash_{\mathsf{u}} A$ (where $\forall/\exists$ are interpreted uniformly), we define a new translation

$$d : (A_1, \ldots, A_n \vdash B) \quad \mapsto \quad d^\circ$$

where the λ-term d° only depends on the proof variables $z_{A_1}, \ldots, z_{A_n}$

- The rules for quantifiers are now translated as follows:

$$\left(\dfrac{\begin{array}{c}\vdots\ d\\ \Gamma \vdash A\end{array}}{\Gamma \vdash \forall x\, A}\right)^\circ := d^\circ \qquad \left(\dfrac{\begin{array}{c}\vdots\ d\\ \Gamma \vdash \forall x\, A\end{array}}{\Gamma \vdash A[x := e]}\right)^\circ := d^\circ$$

$$\left(\dfrac{\begin{array}{c}\vdots\ d\\ \Gamma \vdash A[x := e]\end{array}}{\Gamma \vdash \exists x\, A}\right)^\circ := d^\circ \qquad \left(\dfrac{\begin{array}{c}\vdots\ d_1\\ \Gamma \vdash \exists x\, A\end{array} \quad \begin{array}{c}\vdots\ d_2\\ \Gamma, A \vdash B\end{array}}{\Gamma \vdash B}\right)^\circ := \text{let } z = d_1^\circ \text{ in } d_2^\circ$$

(the other cases of the definition are the same as for $d \mapsto d^*$)

- **Remark:** d° does not depend on first-order variables
  $\Rightarrow$ Witnesses are lost

The uniform interpretation of first-order logic (3/3)

- We can now prove the:

Lemma (Adequacy w.r.t. the uniform interpretation)

Let $d : (A_1, \ldots, A_n \vdash B)$ be a derivation in NJ. Then:

- for all valuations $\rho : \mathsf{FOVar} \to \mathbb{N}$,
- for all realizers $t_1 \Vdash_{\mathrm{u}} A_1[\rho], \ldots, t_n \Vdash_{\mathrm{u}} A_n[\rho]$,

we have: $d^\circ[z_1 := t_1, \ldots, z_n := t_n] \Vdash_{\mathrm{u}} B[\rho]$

Note that we do not need to apply the valuation ρ to the λ-term d°, since the latter does not depend on first-order variables

- **Conclusion:** 1st-order int. logic supports both interpretations:

Non-uniform: (Kleene)	$\forall x\, A(x) \approx \prod_{x \in D} A(x)$	$\exists x\, A(x) \approx \sum_{x \in D} A(x)$
Uniform: (without witnesses)	$\forall x\, A(x) \approx \bigcap_{x \in D} A(x)$	$\exists x\, A(x) \approx \bigcup_{x \in D} A(x)$

Uniformity and typing

Quantifiers also have their uniform typing rules (adequate w.r.t. $\Vdash_u$)

- Uniform typing rules for $\forall$:

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall x\, A}\ {\scriptstyle x \notin FV(\Gamma)} \qquad\qquad \frac{\Gamma \vdash t : \forall x\, A}{\Gamma \vdash t : A[x := e]}$$

Note: $\forall$ treated as an infinitary intersection type

- Uniform typing rules for $\exists$:

$$\frac{\Gamma \vdash t : A[x := e]}{\Gamma \vdash t : \exists x\, A} \qquad\qquad \frac{\Gamma \vdash t : (\exists x\, A) \Rightarrow B}{\Gamma \vdash t : \forall x\,(A \Rightarrow B)}\ {\scriptstyle x \notin FV(B)}$$

Note: $\exists$ treated as an infinitary union type.
Equivalently, its elimination rule can be replaced by:

- A left-rule of the form: $$\frac{\Gamma,\ z : A,\ \Gamma' \vdash t : B}{\Gamma,\ z : \exists x\, A,\ \Gamma' \vdash t : B}\ {\scriptstyle x \notin FV(\Gamma, \Gamma', B)}$$
- A subtyping rule: $(\exists x\, A) \Rightarrow B \ \leq\ \forall x\,(A \Rightarrow B)$ (if $x \notin FV(B)$)

Uniformly realizing the axioms of HA

Lemma (Uniformly realizing true Π^0_1-formulas)

Let $e_1(\vec{x})$, $e_2(\vec{x})$ be FO-terms depending on free variables $\vec{x}$.

If $\mathbb{N} \models \forall\vec{x}\,(e_1(\vec{x}) = e_2(\vec{x}))$, then $0 \Vdash_u \forall\vec{x}\,(e_1(\vec{x}) = e_2(\vec{x}))$

Since all defining equations of function symbols are Π^0_1:

Corollary

All defining equations of function symbols are uniformly realized

Lemma (Uniformly realizing Peano axioms, recall)

$$\lambda z\,.\,z \Vdash_u \forall x\,\forall y\,(s(x) = s(y) \Rightarrow x = y)$$

$$\text{any\_term} \Vdash_u \forall x\,(s(x) \neq 0)$$

What about the induction principle?

Why induction is **not** uniformly realized... (1/2)

Write $A(x) \;:\equiv\; x = 0 \vee \exists y\,(x = s(y))$ ("x is either zero or a successor")

Proposition

We have HA $\vdash \forall x\, A(x)$ but $\not\Vdash_{\mathrm{u}} \forall x\, A(x)$

Proof. HA $\vdash \forall x\, A(x)$ by induction, using the induction predicate $A(x)$.

Let us now assume that $t \Vdash_{\mathrm{u}} \forall x\, A(x)$ for some t, that is: $t \Vdash_{\mathrm{u}} A(n)$ for all $n \in \mathbb{N}$.
For each $n \in \mathbb{N}$, we have $t \succ^* \langle \bar{p}_n, u_n \rangle$ for some $p_n \in \mathbb{N}$ and $u_n \in \Lambda$ such that:

- (Left-hand side of the disjunction) either $p_n = 0$ and $u_n \Vdash n = 0$;
- (Right-hand side of the disjunction) either $p_n = 1$ and $u_n \Vdash \exists y\,(n = s(y))$.

When $n = 0$, the second case is impossible, hence $p_0 = 0$.
And when $n = 1$, the first case is impossible, hence $p_1 = 1$.
From the confluence of $\succ$, we deduce that $p_0 = 0 = p_1 = 1$: contradiction! □

Remark: The proof that $\not\Vdash_{\mathrm{u}} \forall x\, A(x)$ crucially relies on the confluence of $\succ$.
Indeed, if we add a constant $\pitchfork$ (non-deterministic choice) with the rules

$$\pitchfork\, t\, u \succ t \qquad \text{and} \qquad \pitchfork\, t\, u \succ u \qquad \text{(for all terms } t,\, u\text{)}$$

then we easily check that $\pitchfork \langle \bar{0}, \bar{0} \rangle \langle \bar{1}, \bar{0} \rangle \Vdash_{\mathrm{u}} \forall x\, A(x)$

Why induction is **not** uniformly realized... (2/2)

Write $A(x) \;:\equiv\; x = 0 \lor \exists y\,(x = s(y))$ ("x is either zero or a successor")

Corollary (An induction axiom that is not uniformly realized)

$\not\Vdash_{\mathrm{u}}\; A(0) \land \forall x\,(A(x) \Rightarrow A(s(x))) \Rightarrow \forall x\, A(x)$ (with $A(x)$ defined as above)

Proof. Assuming that $t \Vdash_{\mathrm{u}} A(0) \land \forall x\,(A(x) \Rightarrow A(s(x))) \Rightarrow \forall x\, A(x)$ for some t, we easily deduce that $t\,\langle\langle \bar{0}, \bar{0}\rangle,\, \lambda\_\,.\,\langle \bar{1}, \bar{0}\rangle\rangle \Vdash_{\mathrm{u}} \forall x\, A(x)$: contradiction! □

Exercise: Assuming the presence of a non-deterministic choice operator $\pitchfork$ in the language of realizers (cf previous slide):

1. Define a term t_{nat} such that $t_{\mathrm{nat}} \succ^* \bar{n}$ for all $n \in \mathbb{N}$
2. Deduce a term t_{ind} that uniformy realizes all induction axioms
3. More generally, construct a "universal realizer" $\mathbf{t}$ (using $\pitchfork$) such that for each closed formula A:

$$\mathbf{t} \Vdash_{\mathrm{u}} A \quad\text{iff}\quad \mathbf{t} \Vdash A \quad\text{iff}\quad \mathbb{N} \models A$$

Conclusion: (uniform) realizability with $\pitchfork$ $=$ Tarski (in $\mathbb{N}$)

... and how to recover it! (1/2)

To make the induction principle compatible with uniform realizability, we need to go back to Peano's seminal presentation:

Giuseppe Peano. *Arithmetices principia, nova methodo exposita*. 1889

Axiomata.

1. $1 \in \mathrm{N}.$
2. $a \in \mathrm{N} \,.\supset .\, a = a.$
3. $a, b, c \in \mathrm{N} \,.\supset :\, a = b \,.=.\, b = a.$
4. $a, b \in \mathrm{N} \,.\supset \therefore\, a = b \,.\, b = c :\supset .\, a = c.$
5. $a = b \,.\, b \in \mathrm{N} :\supset .\, a \in \mathrm{N}.$
6. $a \in \mathrm{N} \,.\supset .\, a + 1 \in \mathrm{N}.$
7. $a, b \in \mathrm{N}.\supset :\, a = b \,.=.\, a + 1 = b + 1.$
8. $a \in \mathrm{N} \,.\supset .\, a + 1 -= 1.$
9. $k \in \mathrm{K} \therefore 1 \in k \therefore x \in \mathrm{N} \,.\, x \in k :\supset_x .\, x + 1 \in k :: \supset .\, \mathrm{N} \supset k.$

... and how to recover it! (2/2)

To make the induction principle compatible with uniform realizability, we need to go back to Peano's seminal presentation:

Giuseppe Peano. *Arithmetices principia, nova methodo exposita*. 1889

Peano's axioms for arithmetic, using modern notations

1. $1 \in \mathbb{N}$
2. $a \in \mathbb{N} \Rightarrow a = a$
3. $a, b \in \mathbb{N} \Rightarrow (a = b \Leftrightarrow b = a)$
4. $a, b, c \in \mathbb{N} \Rightarrow (a = b \wedge b = c \Rightarrow a = c)$
5. $a = b \wedge b \in \mathbb{N} \Rightarrow a \in \mathbb{N}$
6. $a \in \mathbb{N} \Rightarrow a + 1 \in \mathbb{N}$
7. $a, b \in \mathbb{N} \Rightarrow (a = b \Leftrightarrow a + 1 = b + 1)$
8. $a \in \mathbb{N} \Rightarrow a + 1 \neq 1$
9. $k \in K \wedge 1 \in k \wedge \forall x\, (x \in \mathbb{N} \wedge x \in k \Rightarrow x + 1 \in k) \Rightarrow \mathbb{N} \subseteq k$

(where K is the class of all classes)

Open world assumption: $\mathbb{N}$ is only a subclass of the universe

Peano relative arithmetic (1/2)

- To formalize Peano's open world assumption, we introduce a new first-order theory: Peano relative arithmetic ($PA^{\mathbb{N}}$)

  (As usual, we write $HA^{\mathbb{N}}$ the intuitionistic fragment of $PA^{\mathbb{N}}$)

- The language of $PA^{\mathbb{N}}/HA^{\mathbb{N}}$ is the language of PA/HA enriched with a unary predicate symbol $x \in \mathbb{N}$ ("x is a natural number")

Language of $PA^{\mathbb{N}}/HA^{\mathbb{N}}$

FO-terms $\quad e, e_1 \; ::= \; x \mid f(e_1, \dots, e_k) \qquad$ (f of arity k)

Formulas $\quad A, B \; ::= \; e_1 = e_2 \mid e \in \mathbb{N} \mid \top \mid \bot \mid A \Rightarrow B \mid A \wedge B \mid A \vee B \mid \forall x\, A \mid \exists x\, A$

(Assuming one function symbol f for each definition of a prim. rec. function)

- **Notations:**

$$(\forall x \in \mathbb{N}) A(x) \;:\equiv\; \forall x\, (x \in \mathbb{N} \Rightarrow A(x))$$
$$(\exists x \in \mathbb{N}) A(x) \;:\equiv\; \exists x\, (x \in \mathbb{N} \wedge A(x))$$

Peano relative arithmetic (2/2)

Axioms of $PA^{\mathbb{N}}/HA^{\mathbb{N}}$

Domain of zero and successor

- $0 \in \mathbb{N}$
- $(\forall x \in \mathbb{N})(s(x) \in \mathbb{N})$

Defining equations of all primitive recursive functions in $\mathbb{N}$

- $(\forall x \in \mathbb{N})(x + 0 = x)$, $\quad (\forall x, y \in \mathbb{N})(x + s(y) = s(x + y))$
- $(\forall x \in \mathbb{N})(x \times 0 = 0)$, $\quad (\forall x, y \in \mathbb{N})(x \times s(y) = x \times y + x)$ (etc.)

Peano axioms, relativized to $\mathbb{N}$

- $(\forall x, y \in \mathbb{N})(s(x) = s(y) \Rightarrow x = y)$
- $(\forall x \in \mathbb{N})(s(x) \neq 0)$
- $\forall \vec{z}\,[A(\vec{z}, 0) \wedge (\forall x \in \mathbb{N})(A(\vec{z}, x) \Rightarrow A(\vec{z}, s(x))) \Rightarrow (\forall x \in \mathbb{N})A(\vec{z}, x)]$

From the above axioms, we easily prove that:

Theorem

For each 1st-order term $e(\vec{x})$: $\quad HA^{\mathbb{N}} \vdash (\forall \vec{x} \in \mathbb{N})(e(\vec{x}) \in \mathbb{N})$

Relating PA/HA and $\mathrm{PA}^{\mathbb{N}}/\mathrm{HA}^{\mathbb{N}}$ (1/2)

- The relationship between PA/HA and $\mathrm{PA}^{\mathbb{N}}/\mathrm{HA}^{\mathbb{N}}$ can be studied via a translation $A \mapsto A^{\mathbb{N}} \; : \; \mathscr{L}_{\mathrm{PA}} \to \mathscr{L}_{\mathrm{PA}^{\mathbb{N}}}$ (relativization to $\mathbb{N}$)

Definition of the translation $A \mapsto A^{\mathbb{N}}$

$$\begin{array}{rcl} \top^{\mathbb{N}} & :\equiv & \top \\ (e_1 = e_2)^{\mathbb{N}} & :\equiv & e_1 = e_2 \\ (A \wedge B)^{\mathbb{N}} & :\equiv & A^{\mathbb{N}} \wedge B^{\mathbb{N}} \\ (\forall x\, A)^{\mathbb{N}} & :\equiv & (\forall x \in \mathbb{N}) A^{\mathbb{N}} \end{array} \qquad \begin{array}{rcl} \bot^{\mathbb{N}} & :\equiv & \bot \\ (A \Rightarrow B)^{\mathbb{N}} & :\equiv & A^{\mathbb{N}} \Rightarrow B^{\mathbb{N}} \\ (A \vee B)^{\mathbb{N}} & :\equiv & A^{\mathbb{N}} \vee B^{\mathbb{N}} \\ (\exists x\, A)^{\mathbb{N}} & :\equiv & (\exists x \in \mathbb{N}) A^{\mathbb{N}} \end{array}$$

Theorem

For each $A \in \mathscr{L}_{\mathrm{PA}}$ (closed):

$$\mathrm{PA} \vdash A \quad \text{iff} \quad \mathrm{PA}^{\mathbb{N}} \vdash A^{\mathbb{N}}$$

$$\mathrm{HA} \vdash A \quad \text{iff} \quad \mathrm{HA}^{\mathbb{N}} \vdash A^{\mathbb{N}}$$

- Therefore, the theories PA, $\mathrm{PA}^{\mathbb{N}}$, HA and $\mathrm{HA}^{\mathbb{N}}$ are equiconsistent:

$$\mathrm{PA}^{\mathbb{N}} \approx \underbrace{\mathrm{PA} \approx \mathrm{HA}}_{\text{by inclusion and negative translation}} \approx \mathrm{HA}^{\mathbb{N}}$$

Relating PA/HA and $PA^{\mathbb{N}}/HA^{\mathbb{N}}$ (2/2)

Proof of the equivalence: PA $\vdash A$ iff $\mathbf{PA}^{\mathbb{N}} \vdash A^{\mathbb{N}}$.

(Direct implication) We successively prove that

(1) For all $\Gamma, A \in \mathscr{L}_{PA}$: $\Gamma \vdash_{NK} A$ implies $\Gamma^{\mathbb{N}}, \vec{x} \in \mathbb{N} \vdash_{NK} A^{\mathbb{N}}$, writing $\vec{x} = FV(\Gamma, A)$. (Proof: by induction on the derivation.)

(2) For each axiom A of PA, we have: $HA^{\mathbb{N}} \vdash A^{\mathbb{N}}$

The desired implication immediately follows from (1) and (2).

(Converse implication) For each formula A of $PA^{\mathbb{N}}$, we write $A^{-\mathbb{N}}$ the formula of PA obtained by replacing in A all subformulas of the form $e \in \mathbb{N}$ by the trivial formula $\top$ (thus removing relativizations). Then we prove that

(1) For all $\Gamma, A \in \mathscr{L}_{PA^{\mathbb{N}}}$: $\Gamma \vdash_{NK} A$ implies $\Gamma^{-\mathbb{N}} \vdash_{NK} A^{-\mathbb{N}}$ (Proof: by induction on the derivation.)

(2) For each axiom A of $PA^{\mathbb{N}}$, we have: $HA \vdash A^{-\mathbb{N}}$

(3) For each closed formula $A \in \mathscr{L}_{PA}$: $\vdash_{NJ} (A^{\mathbb{N}})^{-\mathbb{N}} \Leftrightarrow A$ (Proof: by induction on A)

Finally, assuming that $PA^{\mathbb{N}} \vdash A^{\mathbb{N}}$, we deduce that $PA \vdash (A^{\mathbb{N}})^{-\mathbb{N}}$ (from (1) and (2)), and conclude that $PA \vdash A$ (from (3)).

The corresponding equivalence for $HA/HA^{\mathbb{N}}$ is proved similarly. □

$\mathrm{HA}^{\mathbb{N}}$ and uniform realizability (1/2)

- We extend the relation $t \Vdash_u A$ to the new predicate $x \in \mathbb{N}$:

$$t \Vdash_u e \in \mathbb{N} \quad :\equiv \quad t \succ^* \overline{e^{\mathbb{N}}} \qquad \text{(= the value of } e \text{ as a } \lambda\text{-term)}$$

- We observe that:

$$\begin{aligned} t \Vdash_u \underbrace{\forall x\,(x \in \mathbb{N} \Rightarrow A(x))}_{\text{Relativized } \forall} \quad &\text{iff} \quad \forall n\ (t \Vdash_u n \in \mathbb{N} \Rightarrow A(n)) \\ &\text{iff} \quad \forall n\,\forall u\ (u \succ^* \bar{n} \ \Rightarrow\ t\,u \Vdash_u A(n)) \\ &\text{iff} \quad \underbrace{\forall n\ (t\,\bar{n} \Vdash_u A(n))}_{\text{Kleene's non-uniform interpretation of } \forall} \end{aligned}$$

$$\begin{aligned} t \Vdash_u \underbrace{\exists x\,(x \in \mathbb{N} \wedge A(x))}_{\text{Relativized } \forall} \quad &\text{iff} \quad \exists n\ (t \Vdash_u n \in \mathbb{N} \wedge A(n)) \\ &\text{iff} \quad \exists n\,\exists u\,\exists v\ (t \succ^* \langle u, v\rangle\ \wedge u \succ^* \bar{n} \wedge\ v \Vdash_u A(n)) \\ &\text{iff} \quad \underbrace{\exists n\,\exists v\ (t \succ^* \langle \bar{n}, v\rangle\ \wedge\ v \Vdash_u A(n))}_{\text{Kleene's non-uniform interpretation of } \exists} \end{aligned}$$

- **Conclusion:** Relativized uniform $\forall/\exists$ $=$ Non-uniform $\forall/\exists$

$HA^{\mathbb{N}}$ and uniform realizability (2/2)

Lemma (Uniformly realizing the axioms of $HA^{\mathbb{N}}$)

All the axioms of $HA^{\mathbb{N}}$ are uniformly realized:

$$
\begin{array}{rcll}
0 & \Vdash_{u} & 0 \in \mathbb{N} & \\
\lambda x\,.\,\mathtt{S}(x) & \Vdash_{u} & (\forall x \in \mathbb{N}) s(x) \in \mathbb{N} & \\
\lambda \_\,.\,0 & \Vdash_{u} & (\forall x \in \mathbb{N})(x + 0 = 0) & \\
\lambda \_, \_\,.\,0 & \Vdash_{u} & (\forall x, y \in \mathbb{N})(x + s(y) = s(x + y)) & \text{(etc. for each } f\text{)} \\
 & \vdots & & \\
\lambda x, y, z\,.\,z & \Vdash_{u} & (\forall x \in \mathbb{N})(s(x) = s(y) \Rightarrow x = y) & \\
\text{any\_term} & \Vdash_{u} & (\forall x \in \mathbb{N})(s(x) \neq 0) & \\
\mathtt{rec} & \Vdash_{u} & \forall \vec{y}\,[A(\vec{y}, 0) \Rightarrow (\forall x \in \mathbb{N})(A(\vec{y}, x) \Rightarrow A(\vec{y}, s(x))) \Rightarrow (\forall x \in \mathbb{N}) A(\vec{y}, x)] &
\end{array}
$$

(writing $\mathtt{rec} \;:\equiv\; \lambda z_0, z_1, x\,.\,\mathtt{rec}(z_0, z_1, x)$)

Proof: Exercise

Therefore, all theorems of $HA^{\mathbb{N}}$ are uniformly realized:

Theorem (Soundness): If $HA^{\mathbb{N}} \vdash A$, then $t \Vdash_{u} A$ for some t

Kleene realizability vs uniform realizability

Remark: The uniform realizers of the axioms of $PA^{\mathbb{N}}$ are essentially the same as the Kleene realizers of the axioms of PA.

This is due to the following result:

Proposition (Kleene realizability vs uniform realizability)

For all closed formulas A of HA and for all closed λ-terms t:

$$t \Vdash A \qquad \text{iff} \qquad t \Vdash_{\mathsf{u}} A^{\mathbb{N}}$$

Proof. By induction on the size of A (Exercise)

Conclusion: Kleene realiz. $=$ Uniform realiz. $\circ$ $(A \mapsto A^{\mathbb{N}})$

Moreover, the following diagram commutes: (Exercise)

$$\begin{array}{ccc} d : (\Gamma \vdash_{NJ} A) & \xrightarrow{(\_)^{\mathbb{N}}} & d^{\mathbb{N}} : (\Gamma^{\mathbb{N}}, \vec{x} \in \mathbb{N} \vdash_{NJ} A^{\mathbb{N}}) \\ \downarrow (\_)^* & & \downarrow (\_)^\circ \\ d^* & = & (d^{\mathbb{N}})^\circ \end{array}$$

(where $\vec{x} = FV(\Gamma, A)$)

Conclusion

- The equivalence $t \Vdash A$ iff $t \Vdash_{\mathrm{u}} A^{\mathbb{N}}$ implies that:

For all $A \in \mathscr{L}_{\mathrm{PA}}$ and $t \in \Lambda$ (closed):

$$t \Vdash \forall x\, A(x) \quad \text{iff} \quad t \Vdash_{\mathrm{u}} \forall x\,(x \in \mathbb{N} \Rightarrow A^{\mathbb{N}}(x))$$

$$t \Vdash \exists x\, A(x) \quad \text{iff} \quad t \Vdash_{\mathrm{u}} \exists x\,(x \in \mathbb{N} \wedge A^{\mathbb{N}}(x))$$

- **Conclusion:** Non-uniform quant. = relativized uniform quant.:

$$\text{(non-uniform) } \forall x\, A(x) \quad = \quad \text{(uniform) } \forall x\, \underbrace{(D(x) \Rightarrow A(x))}_{\text{type of functions}}$$

$$\text{(non-uniform) } \exists x\, A(x) \quad = \quad \text{(uniform) } \exists x\, \underbrace{(D(x) \wedge A(x))}_{\text{type of pairs}}$$

where $D(x)$ is the domain of quantification

- Uniform realizabilty appears to be more primitive than Kleene's

  $\Rightarrow$ In what follows, we shall sistematically use uniform realizability (while introducing the needed relativization predicates)

A SKELETON in the (intuitionistic) closet... (1/3)

It is well-known that the following equivalences hold in NJ/NK

$$\forall x\,(A(x) \land B(x)) \;\Leftrightarrow\; \forall x\,A(x) \,\land\, \forall x\,B(x) \qquad \text{(Commutation } \forall/\land)$$

$$\exists x\,(A(x) \lor B(x)) \;\Leftrightarrow\; \exists x\,A(x) \,\lor\, \exists x\,B(x) \qquad \text{(Commutation } \exists/\lor)$$

whereas in LJ/LK, we only have the implications

$$\forall x\,(A(x) \lor B(x)) \;\Leftarrow\; \forall x\,A(x) \,\lor\, \forall x\,B(x) \qquad (\forall/\lor \Leftarrow \lor/\forall)$$

$$\exists x\,(A(x) \land B(x)) \;\Rightarrow\; \exists x\,A(x) \,\land\, \forall x\,B(x) \qquad (\exists/\land \Rightarrow \land/\exists)$$

The converse implications do not hold... Really?

Proposition (The 'scandalous commutation' $\forall/\lor$)

Given formulas $A(x)$ and $B(x)$ depending only on x, we have:

$$\langle \lambda z\,.\,z,\ \lambda z\,.\,z \rangle \;\Vdash_{\mathrm{u}}\; \forall x\,(A(x) \lor B(x)) \Leftrightarrow \forall x\,A(x) \,\lor\, \forall x\,B(x)$$

Proof. Just check that both sides of $\Leftrightarrow$ have the same uniform realizers. □

Note: The dual commutation $\exists/\land$ is not uniformly realizable

A SKELETON in the (intuitionistic) closet... (2/3)

The 'scandalous commutation' $\forall/\vee$ also holds in all (parametrically) polymorphic functional languages (ML, Haskell), where both types

$$\forall\alpha.(\tau(\alpha)+\sigma(\alpha)) \qquad \text{and} \qquad (\forall\alpha.\tau(\alpha))+(\forall\alpha.\sigma(\alpha))$$

have (at least morally) the same inhabitants

Nevertheless, we can observe that:

1. In classical logic, the commutation $\forall/\vee$ trivializes the universe:

Proposition: $\mathrm{LK}+\mathrm{comm}(\forall/\vee)\ \vdash\ \forall x\,\forall y\,(x=y)$

Proof. Classically, we have:

	$\forall x\,\forall y\,(x=y\vee x\neq y)$	(by excluded middle)
hence	$\forall x\,(\forall y\,(x=y)\vee\forall y\,(x\neq y))$	(by $\mathrm{comm}(\forall/\vee)$)
and since	$\neg\forall y\,(x\neq y)$	(take $y=x$ as a counter example)
we get:	$\forall x\,\forall y\,(x=y)$	

□

So that all non-trivial classical theories (PA, ZF, ...) refute the commutation $\forall/\vee$

A SKELETON in the (intuitionistic) closet... (3/3)

2. The commutation $\forall/\vee$ is compatible with $\mathsf{HA}^{\mathbb{N}}$

Proposition: The theory $\mathsf{HA}^{\mathbb{N}} + \mathrm{comm}(\forall/\vee)$ is consistent

Proof. All axioms of $\mathsf{HA}^{\mathbb{N}}$ are universally realized, as well as $\mathrm{comm}(\forall/\vee)$. □

3. The commutation $\forall/\vee$ is incompatible with HA

Proposition: The theory $\mathsf{HA} + \mathrm{comm}(\forall/\vee)$ is inconsistent

Proof. We observe that:

	$\mathsf{HA} \vdash \forall x\,(x = 0 \vee \exists y\,(x = s(y)))$
hence	$\mathsf{HA} + \mathrm{comm}(\forall/\vee) \vdash \forall x\,(x = 0) \vee \forall x\,\exists y\,(x = s(y)))$
But since	$\mathsf{HA} \vdash \neg\forall x\,(x = 0)$
and since	$\mathsf{HA} \vdash \neg\forall x\,\exists y\,(x = s(y))$
we get:	$\mathsf{HA} + \mathrm{comm}(\forall/\vee) \vdash \bot$

□

Remark: The commutation $\forall/\vee$ remains compatible with all intuitionistic theories where quantifiers can be interpreted uniformly: $\mathsf{HA}^{\mathbb{N}}$, IZ, IZF (etc.)

Plan

1. Kleene realizability
2. Gödel-Gentzen negative translation
3. Uniformity and relativization
4. Lafont-Reus-Streicher negative translation

Kleene realizability and negative translations (recall, 1/2)

Recall: Classical proofs can be turned into programs, by composing Kleene realizability with a negative translation. For example:

Gödel-Genzten translation $A \mapsto A^{\mathbf{G}}$ (Recall)

$$
\begin{array}{rclrcl}
\top^{\mathbf{G}} & :\equiv & \top & \bot^{\mathbf{G}} & :\equiv & \bot \\
(A \Rightarrow B)^{\mathbf{G}} & :\equiv & A^{\mathbf{G}} \Rightarrow B^{\mathbf{G}} & (e_1 = e_2)^{\mathbf{G}} & :\equiv & \neg\neg(e_1 = e_2) \\
(A \wedge B)^{\mathbf{G}} & :\equiv & A^{\mathbf{G}} \wedge B^{\mathbf{G}} & (A \vee B)^{\mathbf{G}} & :\equiv & \neg(\neg A^{\mathbf{G}} \wedge \neg B^{\mathbf{G}}) \\
(\forall x\, A)^{\mathbf{G}} & :\equiv & \forall x\, A^{\mathbf{G}} & (\exists x\, A)^{\mathbf{G}} & :\equiv & \neg\forall x\, \neg A^{\mathbf{G}}
\end{array}
$$

Theorem (Soundness)

1. $\mathrm{LK} \vdash A^{\mathbf{G}} \Leftrightarrow A$
2. If $d : (\mathrm{PA} \vdash A)$, then $d^{\mathbf{G}} : (\mathrm{HA} \vdash A^{\mathbf{G}})$

- **Problem:** $A^{\mathbf{G}}$ is always Harrop; therefore:
  - Extracted λ-term $(d^{\mathbf{G}})^*$ has no computational contents
  - Kleene $\circ$ $(A \mapsto A^{\mathbf{G}})$ mimics Tarski: $\Vdash A^{\mathbf{G}}$ iff $\mathbb{N} \models A$

Kleene realizability and negative translations (recall, 2/2)

Friedman's R-translation $A \mapsto A^{\mathbf{F}}$ (Recall)

$$\top^{\mathbf{F}} :\equiv \top \qquad \bot^{\mathbf{F}} :\equiv R$$
$$(A \Rightarrow B)^{\mathbf{F}} :\equiv A^{\mathbf{F}} \Rightarrow B^{\mathbf{F}} \qquad (e_1 = e_2)^{\mathbf{F}} :\equiv \neg_R\neg_R(e_1 = e_2)$$
$$(A \wedge B)^{\mathbf{F}} :\equiv A^{\mathbf{F}} \wedge B^{\mathbf{F}} \qquad (A \vee B)^{\mathbf{F}} :\equiv \neg_R(\neg_R A^{\mathbf{F}} \wedge \neg_R B^{\mathbf{F}})$$
$$(\forall x\, A)^{\mathbf{F}} :\equiv \forall x\, A^{\mathbf{F}} \ \ (\text{if } x \notin FV(R)) \qquad (\exists x\, A)^{\mathbf{F}} :\equiv \neg_R \forall x\, \neg_R A^{\mathbf{F}} \ \ (\text{if } x \notin FV(R))$$

writing: $\neg_R A :\equiv A \Rightarrow R$, where R is the return formula

Theorem (Soundness & Π^0_1-conservativity)

1. If $d : (\mathrm{PA} \vdash A)$, then $d^{\mathbf{F}} : (\mathrm{HA} \vdash A^{\mathbf{F}})$ (for any return formula R)
2. Given $A \equiv \forall x\, \exists y\, f(x, y) = 0$: ($\Pi^0_1$-formula)
   If $d : (\mathrm{PA} \vdash A)$, then F-trick($d^{\mathbf{G}}$) $: (\mathrm{HA} \vdash A)$ (using a suitable R)

- **Pro:** In the Π^0_1-case, the program (F-trick($d^{\mathbf{F}}$))* does the expected job
- **Contra:** The translation $d \mapsto d^{\mathbf{F}}$ completely changes the structure of the underlying proof. **Possible fix:** cf next slides

The Lafont-Reus-Streicher negative translation (1/2)

The Lafont-Reus-Streicher (LRS) translation works across two languages:

- **Source language:** A minimal language for classical logic:

Formulas $A, B \quad ::= \quad p(e_1, \ldots, e_k) \quad | \quad \bot \quad | \quad A \Rightarrow B \quad | \quad \forall x\, A$

(no equality, no arithmetic – remaining constructions defined by De Morgan laws)

(+ deduction rules of LK)

- **Target language:** The usual language of LJ

Principle of the LRS-translation: Translate each formula A (of the source language) into two formulas (of the target language):

- A formula $A^{\bot}$ (target language) representing the negation of A
- A formula $A^{\mathbf{LRS}}$ (target language) representing A itself

Moreover, $A^{\mathbf{LRS}}$ is uniformly defined by $A^{\mathbf{LRS}} \;:\equiv\; \neg_R A^{\bot} \;\equiv\; A^{\bot} \Rightarrow R$, where R is the return formula that parameterizes the construction

The Lafont-Reus-Streicher negative translation (2/2)

- To every predicate symbol p (source language) we associate a predicate symbol $\bar{p}$ (target language) representing the negation of p
- The translations $A \mapsto A^\perp$ and $A \mapsto A^{\mathbf{LRS}}$ (source $\to$ target) are defined by mutual recursion as follows:

$$
\begin{array}{rclcrcl}
(p(e_1,\ldots,e_k))^\perp & :\equiv & \bar{p}(e_1,\ldots,e_k) & \qquad & \perp^\perp & :\equiv & \top \\
(A \Rightarrow B)^\perp & :\equiv & A^{\mathbf{LRS}} \land B^\perp & \qquad & (\forall x\, A)^\perp & :\equiv & \exists x\, A^\perp
\end{array}
$$

$$A^{\mathbf{LRS}} \;:\equiv\; \neg_R A^\perp \;\equiv\; A^\perp \Rightarrow R$$

Theorem (Soundness)

(1) When $R \equiv \perp$, and under the axioms $\forall \vec{x}\,(p(\vec{x}) \Leftrightarrow \bar{p}(\vec{x}))$ (for all $p, \bar{p}$)

$$\mathrm{LK} + \text{axioms} \vdash A^\perp \Leftrightarrow \neg A \qquad \text{and} \qquad \mathrm{LK} + \text{axioms} \vdash A^{\mathbf{LRS}} \Leftrightarrow A$$

(2) If $\mathrm{LK} \vdash A$, then $\mathrm{LJ} \vdash A^{\mathbf{LRS}}$ (independently from the formula R)

Proof: (1) By induction on A
(2) By induction on the derivation (Exercise)

Computational interpretation

- **Intuition:** The translated formula $A^\perp$ represents the type of stacks opposing (classical) terms of type A:

$$(A_1 \Rightarrow \cdots \Rightarrow A_n \Rightarrow B)^\perp \quad \equiv \quad A_1^{\mathbf{LRS}} \wedge \cdots \wedge A_n^{\mathbf{LRS}} \wedge B^\perp$$

$$(A_1 \to \cdots \to A_n \to B)^\perp \quad \equiv \quad A_1^{\mathbf{LRS}} \times \cdots \times A_n^{\mathbf{LRS}} \times B^\perp$$

- To analyze the computational contents of the LRS-translation, we now need to work across two λ-calculi:
    - A source calculus to represent classical proofs:

      $$\lambda_{\text{source}} \;=\; \lambda_\to + \text{œ} : ((A \to B) \to A) \to A \qquad \text{(Peirce's law)}$$

      (Polymorphic constant œ introduces classical reasoning)
    - An intuitionistic target calculus to represent translated proofs:

      $$\lambda_{\text{target}} \;=\; \lambda_{\to,\times}$$

      (In this calculus, pairs are used to represent stacks)

The source λ-calculus ($\{\bot, \Rightarrow, \forall\}$-fragment of LK)

Syntax (Minimal fragment of LK)

Types $A, B \quad ::= \quad \bot \quad | \quad p(e_1, \dots, e_k) \quad | \quad A \Rightarrow B \quad | \quad \forall x\, A$

Proof-terms $t, u \quad ::= \quad z \quad | \quad \lambda z\,.\,t \quad | \quad tu \quad | \quad \mathbf{c\!c}$

- Classical logic obtained by introducing an inert constant $\mathbf{c\!c}$ (call/cc) for Peirce's law (taken as an axiom) $\Rightarrow$ No reduction rule!
- Constructions $\top$, $\wedge$, $\vee$, $\exists$ encoded using De Morgan laws (= full LK)

Typing rules

$$\frac{}{\Gamma \vdash z : A}\ {\scriptstyle (z:A)\in\Gamma} \qquad \frac{}{\Gamma \vdash \mathbf{c\!c} : ((A \Rightarrow B) \Rightarrow A) \Rightarrow A}$$

$$\frac{\Gamma, z : A \vdash t : B}{\Gamma \vdash \lambda z\,.\,t : A \Rightarrow B} \qquad \frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash tu : B}$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall x\, A}\ {\scriptstyle x\notin FV(\Gamma)} \qquad \frac{\Gamma \vdash t : \forall x\, A}{\Gamma \vdash t : A[x := e]} \qquad \frac{\Gamma \vdash t : \bot}{\Gamma \vdash t : A}$$

Note: $\forall$ is treated uniformly: $\forall x\, A(x) \ \approx\ \bigcap_x A(x)$ (no function argument!)

The target λ-calculus

($\{\top, \Rightarrow, \wedge, \exists\}$-fragment of LJ)

Syntax (Fragment of LJ)

Types $\quad A, B \quad ::= \quad \top \mid \bar{p}(e_1, \ldots, e_k) \mid A \Rightarrow B \mid A \wedge B \mid \exists x\, A$

Proof-terms $\quad t, u \quad ::= \quad z \mid \lambda z\,.\,t \mid tu \mid \langle t, u\rangle \mid \pi_1(t) \mid \pi_2(t)$

$+$ usual reduction rules for proof-terms

Typing rules

$$\frac{}{\Gamma \vdash z : A}\ {\scriptstyle (z:A)\in\Gamma} \qquad \frac{}{\Gamma \vdash t : \top}\ {\scriptstyle FV(t)\subseteq \mathrm{dom}(\Gamma)}$$

$$\frac{\Gamma, z : A \vdash t : B}{\Gamma \vdash \lambda z\,.\,t : A \Rightarrow B} \qquad \frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash tu : B}$$

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash t : B}{\Gamma \vdash \langle t, u\rangle : A \wedge B} \qquad \frac{\Gamma \vdash t : A \wedge B}{\Gamma \vdash \pi_1(t) : A} \qquad \frac{\Gamma \vdash t : A \wedge B}{\Gamma \vdash \pi_2(t) : B}$$

$$\frac{\Gamma \vdash t : A[x := e]}{\Gamma \vdash t : \exists x\, A} \qquad \frac{\Gamma, z : A \vdash t : B}{\Gamma, z : \exists x\, A \vdash t : B}\ {\scriptstyle x\notin FV(\Gamma, B)}$$

Note: $\exists$ treated uniformly: $\exists x\, A(x) \approx \bigcup_x A(x)$ (no witness!)

The Lafont-Reus-Streicher logical translation

- The logical translation $A \mapsto A^{\mathbf{LRS}}$

$$(p(e_1,\dots,e_k))^\perp :\equiv \bar{p}(e_1,\dots,e_k) \qquad \perp^\perp :\equiv \top$$
$$(A \Rightarrow B)^\perp :\equiv A^{\mathbf{LRS}} \wedge B^\perp \qquad (\forall x\, A)^\perp :\equiv \exists x\, A^\perp$$
$$A^{\mathbf{LRS}} :\equiv \neg_R A^\perp$$

corresponds to a program transformation on untyped proof terms, called a continuation-passing style (CPS) translation:

$$(z)^{\mathbf{LRS}} :\equiv \lambda s \,.\, z\, s$$
$$(\lambda z \,.\, t)^{\mathbf{LRS}} :\equiv \lambda \langle z, s_0\rangle \,.\, t^{\mathbf{LRS}}\, s_0$$
$$(tu)^{\mathbf{LRS}} :\equiv \lambda s_0 \,.\, t^{\mathbf{LRS}}\, \langle u^{\mathbf{LRS}}, s_0\rangle$$
$$(\text{cc})^{\mathbf{LRS}} :\equiv \lambda\langle z, s_0\rangle \,.\, z\, \langle \mathrm{k}_{s_0}, s_0\rangle$$
$$\text{where } \mathrm{k}_s :\equiv \lambda\langle z, \_\rangle \,.\, z\, s$$

Note: $\lambda\langle z, s\rangle \,.\, t$ defined as $\lambda z_0 \,.\, (\lambda zs \,.\, t)\, (\pi_1(z_0))\, (\pi_2(z_0))$

Theorem (Soundness)

If $\Gamma \vdash t : A$ (in the source λ-calculus)

then $\Gamma^{\mathbf{LRS}} \vdash t^{\mathbf{LRS}} : A^{\mathbf{LRS}}$ (in the target λ-calculus)

Computational analysis

- Given a term $t : A$ and a "stack" $s : A^{\perp}$ (in the target calculus), we use the notation $t \mathbin{@} s \;:\equiv\; t\,s$ (application of t to the stack s)

- We observe that:

$$
\begin{array}{rcl}
(\lambda z\,.\,t)^{\mathbf{LRS}} \mathbin{@} \langle u, s\rangle & \equiv & (\lambda\langle z, s_0\rangle\,.\,t^{\mathbf{LRS}}\,s_0) \mathbin{@} \langle u, s\rangle \\
& \succ^* & t^{\mathbf{LRS}}[z := u] \mathbin{@} s \\[1em]
(t\,u)^{\mathbf{LRS}} \mathbin{@} s & \equiv & (\lambda s_0\,.\,t^{\mathbf{LRS}}\,\langle u^{\mathbf{LRS}}, s_0\rangle)\,s \\
& \succ^* & t^{\mathbf{LRS}} \mathbin{@} \langle u^{\mathbf{LRS}}, s\rangle \\[1em]
\text{Œ}^{\mathbf{LRS}} \mathbin{@} \langle u, s\rangle & \equiv & (\lambda\langle z, s_0\rangle\,.\,z\,\langle \mathrm{k}_{s_0}, s_0\rangle) \mathbin{@} \langle u, s\rangle \\
& \succ^* & u \mathbin{@} \langle \mathrm{k}_s, s\rangle \\[1em]
\mathrm{k}_s \mathbin{@} \langle u, s'\rangle & \equiv & (\lambda\langle z, \_\rangle\,.\,z\,s) \mathbin{@} \langle u, s'\rangle \\
& \succ^* & u \mathbin{@} s
\end{array}
$$

Towards the Krivine abstract machine

- From the computational behavior of translated proof terms $t^{\mathbf{LRS}}$...

$$
\begin{array}{rcl}
(\lambda z\,.\,t)^{\mathbf{LRS}} \;@\; \langle u, s\rangle & \succ & t^{\mathbf{LRS}}[z := u] \;@\; s \\
(tu)^{\mathbf{LRS}} \;@\; s & \succ & t^{\mathbf{LRS}} \;@\; \langle u^{\mathbf{LRS}}, s\rangle \\
(\text{œ})^{\mathbf{LRS}} \;@\; \langle u, s\rangle & \succ & u \;@\; \langle \mathsf{k}_s, s\rangle \\
\mathsf{k}_s \;@\; \langle u, s'\rangle & \succ & u \;@\; s
\end{array}
$$

... we deduce evaluation rules for classical proof terms:

Krivine Abstract Machine (KAM)

Grab	$\lambda z\,.\,t \star u \cdot \pi$	$\succ$	$t[z := u] \star \pi$
Push	$tu \star \pi$	$\succ$	$t \star u \cdot \pi$
Save	$\text{œ} \star u \cdot \pi$	$\succ$	$u \star \mathsf{k}_\pi \cdot \pi$
Restore	$\mathsf{k}_\pi \star u \cdot \pi'$	$\succ$	$u \star \pi$

- Reformulating Kleene realizability through the LRS translation (and its CPS), we get Krivine classical realizability (cf next talk)