Nonstandard analysis in Krivine realizability

Bruno Dinis
CMAFcIO
University of Lisboa

Étienne MIQUEY
LSV, CNRS
INRIA, ÉNS Paris-Saclay

Days in Logic 2020

Nonstandard analysis

Reaching the inaccessible

Once upon the 17th century...

Leibniz

Sit a quantitas data conſtans, erit da æqualis 0, & d ax erit æqu. a dx: ſi ſit y æqu. v (ſeu ordinata quævis curvæ YY, æqualis cuivis ordinatæ reſpondenti curvæ VV) erit dy æqu. dv. Jam *Additio & Subtractio:* ſi ſit z -y † vv † x æqu. v, erit dz--y † vv † x ſeu dv, æqu. dz--dy † dvv † dx. *Multiplicatio*, dxv æqu. xdv † vdx, ſeu poſito y æqu. xv, fiet dy æqu. xdv † vdx. In arbitrio enim eſt vel formulam,

$$y = x^2$$
$$y + dy = (x + dx)^2$$
$$y + dy = x^2 + 2x\,dx + dx^2$$
$$\frac{dy}{dx} = 2x + dx = 2x$$

Once upon the 17th century...

Leibniz

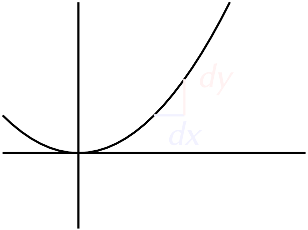
Sit a quantitas data conſtans, erit da æqualis 0, & d ax erit æqu. a dx: ſi ſit y æqu. v (ſeu ordinata quævis curvæ YY, æqualis cuivis ordinatæ reſpondenti curvæ VV) erit dy æqu. dv. Jam *Additio & Subtractio*: ſi ſit z -y † vv † x æqu. v, erit dz--y † vv † x ſeu dv, æqu. dz--dy † dvv † dx. *Multiplicatio*, dxv æqu. xdv † vdx, ſeu poſito y æqu. xv, fiet dy æqu. xdv † vdx. In arbitrio enim eſt vel formulam,

$$y = x^2$$

$$y + dy = (x + dx)^2$$

$$y + dy = x^2 + 2x\,dx + dx^2$$

$$\frac{dy}{dx} = 2x + dx = 2x$$

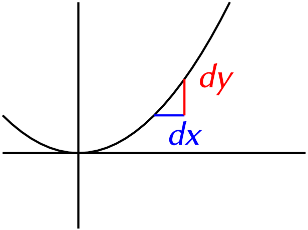


Once upon the 17th century...

Leibniz

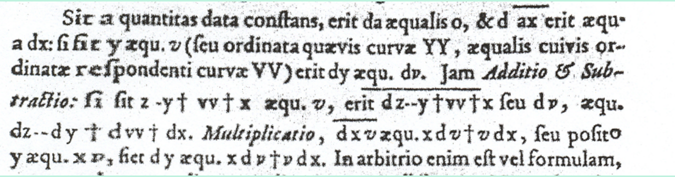
Sit a quantitas data conſtans, erit da æqualis 0, & d ax erit æqu. a dx: ſi ſit y æqu. v (ſeu ordinata quævis curvæ YY, æqualis cuivis ordinatæ reſpondenti curvæ VV) erit dy æqu. dv. Jam *Additio & Subtractio:* ſi ſit z -y † vv † x æqu. v, erit dz--y † vv † x ſeu dv, æqu. dz--dy † dvv † dx. *Multiplicatio*, dxv æqu. xdv † vdx, ſeu poſito y æqu. xv, fiet dy æqu. xdv † vdx. In arbitrio enim eſt vel formulam,

$$y = x^2$$
$$y + dy = (x + dx)^2$$
$$y + dy = x^2 + 2x\,dx + dx^2$$
$$\frac{dy}{dx} = 2x + dx = 2x$$



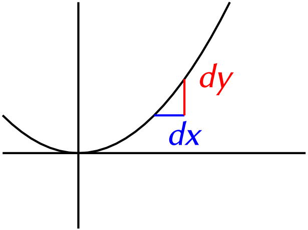
Once upon the 17th century...

Leibniz



Sit a quantitas data constans, erit da æqualis 0, & d ax erit æqu- a dx: si sit y æqu. v (seu ordinata quævis curvæ YY, æqualis cuivis ordinatæ respondenti curvæ VV) erit dy æqu. dv. Jam *Additio & Subtractio:* si sit z -y † vv † x æqu. v, erit d z--y † vv † x seu dv, æqu. dz--dy † dvv † dx. *Multiplicatio*, dxv æqu. xdv † vdx, seu posito y æqu. xv, fiet dy æqu. xdv † vdx. In arbitrio enim est vel formulam,

$$y = x^2$$
$$y + dy = (x + dx)^2$$
$$y + dy = x^2 + 2x dx + dx^2$$
$$\frac{dy}{dx} = 2x + dx = 2x$$



Once upon the 17th century...

Leibniz

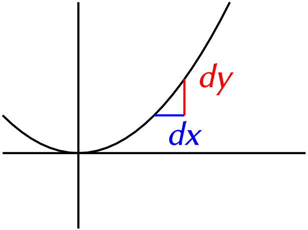
Sit a quantitas data conſtans, erit da æqualis o, & d $\overline{ax}$ erit æqu-
a dx: ſi ſit y æqu. v (ſeu ordinata quævis curvæ YY, æqualis cuivis or-
dinatæ reſpondenti curvæ VV) erit dy æqu. dv. Jam *Additio & Sub-
tractio:* ſi ſit z -y † vv † x æqu. v, erit d $\overline{z\text{-}y \dagger vv \dagger x}$ ſeu dv, æqu.
dz--dy † dvv † dx. *Multiplicatio*, d $\overline{xv}$ æqu. xdv † vdx, ſeu poſito
y æqu. xv, fiet dy æqu. xdv † vdx. In arbitrio enim eſt vel formulam,

$$y = x^2$$

$$y + dy = (x + dx)^2$$

$$y + dy = x^2 + 2x\,dx + dx^2$$

$$\frac{dy}{dx} = 2x + dx = 2x$$



Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Z}^k$: infinite
- $\mathbb{R}$, $\mathcal{F}(\mathbb{R}, \mathbb{R})$, $\mathcal{P}(\mathbb{N})$: (very) infinite

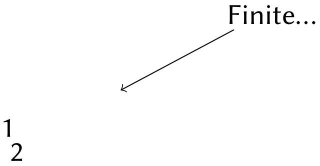
and objects that are **concretely accessible**:

Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Z}^k$: infinite
- $\mathbb{R}$, $\mathcal{F}(\mathbb{R}, \mathbb{R})$, $\mathcal{P}(\mathbb{N})$: (very) infinite

and objects that are **concretely accessible**:



Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Z}^k$: infinite
- $\mathbb{R}$, $\mathcal{F}(\mathbb{R}, \mathbb{R})$, $\mathcal{P}(\mathbb{N})$: (very) infinite

and objects that are **concretely accessible**:

Finite...

1

2

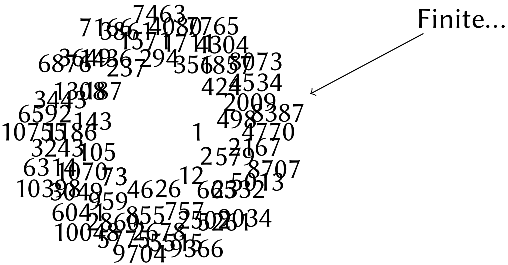
$73_{4626}{}^{12}$

Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Z}^k$: infinite
- $\mathbb{R}$, $\mathcal{F}(\mathbb{R},\mathbb{R})$, $\mathcal{P}(\mathbb{N})$: (very) infinite

and objects that are **concretely accessible**:

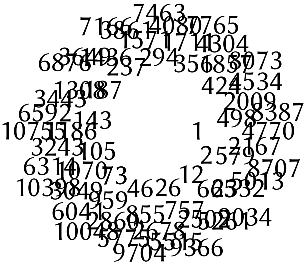


Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}, \mathbb{Z}, \mathbb{Z}^k$: infinite
- $\mathbb{R}, \mathcal{F}(\mathbb{R}, \mathbb{R}), \mathcal{P}(\mathbb{N})$: (very) infinite

and objects that are **concretely accessible**:



Finite...

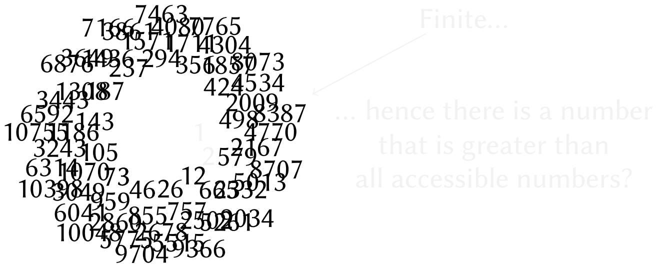
... hence there is a number that is greater than all accessible numbers?

Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}, \mathbb{Z}, \mathbb{Z}^k$: infinite
- $\mathbb{R}, \mathcal{F}(\mathbb{R}, \mathbb{R}), \mathcal{P}(\mathbb{N})$: (very) infinite

and objects that are **concretely accessible**:



Intuition :

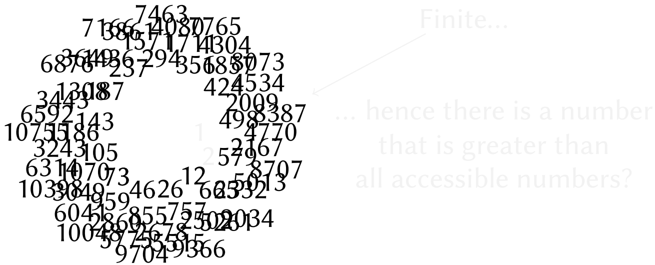
Accessible / non-accessible elements

Intuitive explanation

Mismatch between **theoretical objects**:

- $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Z}^k$: infinite
- $\mathbb{R}$, $\mathcal{F}(\mathbb{R}, \mathbb{R})$, $\mathcal{P}(\mathbb{N})$: (very) infinite

and objects that are **concretely accessible**:



Robinson (1961):

Standard / non-standard elements

Internal set theory

- Internal / external formula
- **Transfer:** $A(x)$ internal

$$\forall^{\mathrm{st}} x. A(x) \Rightarrow \forall x. A(x)$$

- **Idealization:** $R(x, y)$ internal relation

$$\forall^{\mathrm{st}}(n \in \mathbb{N}). \exists^{\mathrm{st}} x. \forall^{\mathrm{st}} y. (y \leq n \Rightarrow R(x, y)) \Rightarrow \exists x. \forall^{\mathrm{st}} y. R(x, y)$$

- **Standardization:** any $C(x)$

$$\forall^{\mathrm{st}} B. \exists^{\mathrm{st}} A. \forall^{\mathrm{st}} z. (z \in A \Leftrightarrow z \in B \wedge C(z))$$

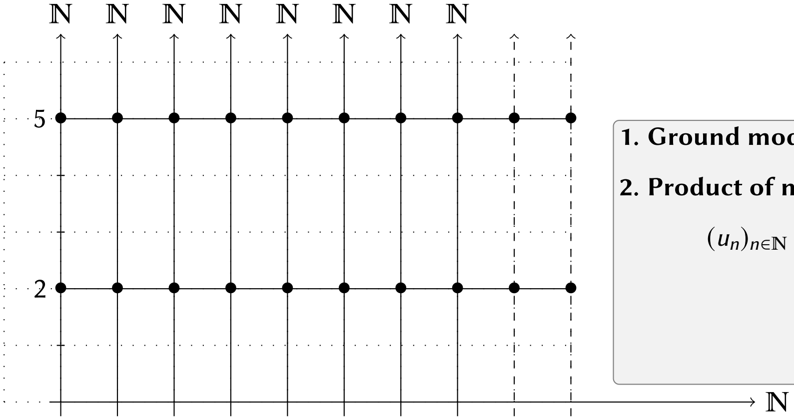
Nonstandard models

1. Ground model

3. Quotient ($w.r.t.\ \mathcal{U}$)

$[u_n]$

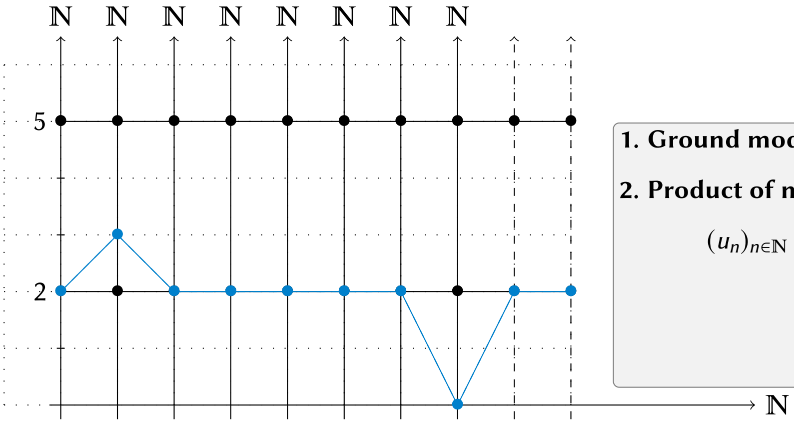
Nonstandard models



1. **Ground model**
2. **Product of models**

$$(u_n)_{n\in\mathbb{N}}$$

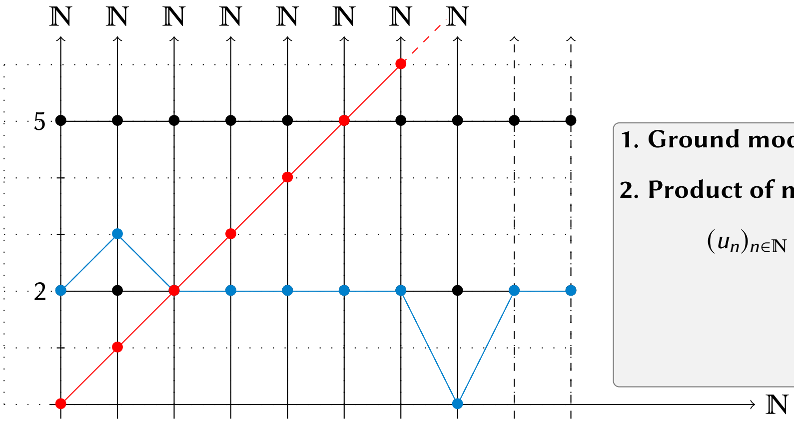
Nonstandard models



1. **Ground model**
2. **Product of models**

$$(u_n)_{n\in\mathbb{N}}$$

Nonstandard models

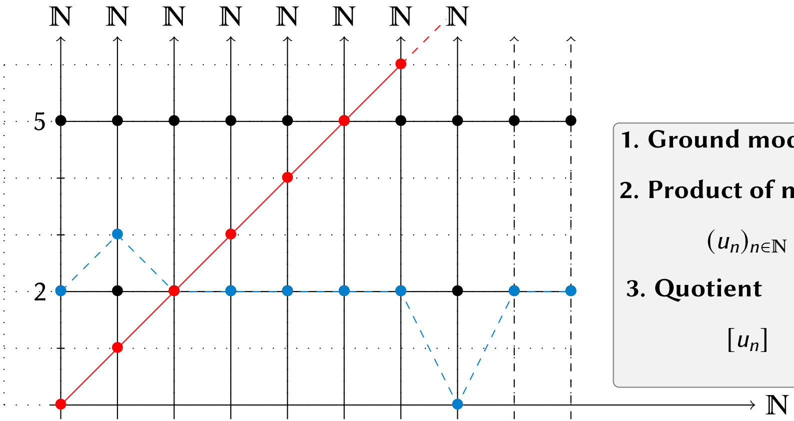


1. Ground model

2. Product of models

$$(u_n)_{n\in\mathbb{N}}$$

Nonstandard models



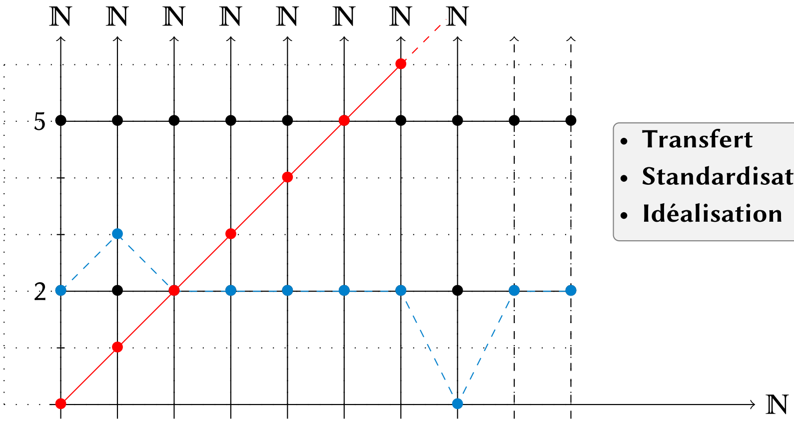
1. **Ground model**
2. **Product of models**

$$(u_n)_{n \in \mathbb{N}}$$

3. **Quotient** (*w.r.t.* $\mathcal{U}$)

$$[u_n]$$

Nonstandard models



- **Transfert**
- **Standardisation**
- **Idéalisation**

Krivine realizability

Unveiling the computational contents of proofs

The λ-calculus

Syntax:

$$t, u \quad ::= \quad x \quad | \quad \lambda x.t \quad | \quad t\, u$$

(variables) $\qquad x \mapsto f(x) \qquad f\,2$

Reduction

$$(\lambda x.t)\, u \longrightarrow_\beta t[u/x]$$

+ contextual closure: $\quad C[t] \longrightarrow_\beta C[t'] \quad (\text{ if } t \longrightarrow_\beta t')$

Examples:

$$(\lambda x.\lambda y.y\, x)\, \bar{2}\, t \longrightarrow_\beta (\lambda y.y\, \bar{2})\, t \longrightarrow_\beta t\, \bar{2}$$

$$\omega = (\lambda x.x\, x)\, (\lambda x.x\, x) \longrightarrow_\beta (\lambda x.x\, x)\, (\lambda x.x\, x) \longrightarrow_\beta \dots$$

Turing completeness

The λ-calculus and Turing machines are equivalent.

The λ-calculus

Syntax:

$$t, u \quad ::= \quad \underset{\text{(variables)}}{x} \quad | \quad \underset{x \mapsto f(x)}{\lambda x.t} \quad | \quad \underset{f\,2}{t\,u}$$

Reduction

$$(\lambda x.t)\,u \longrightarrow_\beta t[u/x]$$

+ contextual closure: $C[t] \longrightarrow_\beta C[t']$ (if $t \longrightarrow_\beta t'$)

Examples:

$$(\lambda x.\lambda y.y\,x)\,\bar{2}\,t \longrightarrow_\beta (\lambda y.y\,\bar{2})\,t \longrightarrow_\beta t\,\bar{2}$$

$$\omega = (\lambda x.x\,x)\,(\lambda x.x\,x) \longrightarrow_\beta (\lambda x.x\,x)\,(\lambda x.x\,x) \longrightarrow_\beta \ldots$$

Turing completeness

The λ-calculus and Turing machines are equivalent.

The λ-calculus

Syntax:

$$t, u \quad ::= \quad x \quad | \quad \lambda x.t \quad | \quad t\, u$$

(variables) $\qquad x \mapsto f(x) \qquad f\,2$

Reduction

$$(\lambda x.t)\, u \longrightarrow_\beta t[u/x]$$

+ contextual closure: $\quad C[t] \longrightarrow_\beta C[t'] \quad$ (if $t \longrightarrow_\beta t'$)

Examples:

$$(\lambda x.\lambda y.y\, x)\, \bar{2}\, t \longrightarrow_\beta (\lambda y.y\, \bar{2})\, t \longrightarrow_\beta t\, \bar{2}$$

$$\omega = (\lambda x.x\, x)\, (\lambda x.x\, x) \longrightarrow_\beta (\lambda x.x\, x)\, (\lambda x.x\, x) \longrightarrow_\beta \dots$$

Turing completeness

The λ-calculus and Turing machines are equivalent.

The λ-calculus

Syntax:

$$t, u \quad ::= \quad x \quad | \quad \lambda x.t \quad | \quad t\,u$$

(variables) $\qquad x \mapsto f(x) \qquad f\,2$

Reduction

$$(\lambda x.t)\,u \longrightarrow_\beta t[u/x]$$

+ contextual closure: $\quad C[t] \longrightarrow_\beta C[t'] \quad$ (if $t \longrightarrow_\beta t'$)

Examples:

$$(\lambda x.\lambda y.y\,x)\,\bar{2}\,t \longrightarrow_\beta (\lambda y.y\,\bar{2})\,t \longrightarrow_\beta t\,\bar{2}$$

$$\omega = (\lambda x.x\,x)\,(\lambda x.x\,x) \longrightarrow_\beta (\lambda x.x\,x)\,(\lambda x.x\,x) \longrightarrow_\beta \dots$$

Turing completeness

The λ-calculus and Turing machines are equivalent.

Types

Goal:

Eliminate unwanted behaviour

Simple types: $A, B \quad ::= \quad \underset{\mathbb{N}}{X} \quad | \quad \underset{\mathbb{R} \to \mathbb{N}}{A \to B}$

Sequent:

Hypothesis $\boxed{\Gamma \vdash t : A}$ Conclusion

Typing rules:

$$\frac{(x : A) \in \Gamma}{\Gamma \vdash x : A}\ (\text{Ax}) \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}\ (\to_I) \qquad \frac{\Gamma \vdash t : A \to B \quad \Gamma \vdash u : A}{\Gamma \vdash t\,u : B}\ (\to_E)$$

Types

Goal:

Eliminate unwanted behaviour

Simple types:

$$A, B \quad ::= \quad \underset{\mathbb{N}}{X} \quad | \quad \underset{\mathbb{R} \to \mathbb{N}}{A \to B}$$

Sequent:

Hypothesis $\boxed{\Gamma \vdash t : A}$ Conclusion

Typing rules:

$$\frac{(x : A) \in \Gamma}{\Gamma \vdash x : A}\ (\mathrm{Ax}) \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}\ (\to_I) \qquad \frac{\Gamma \vdash t : A \to B \quad \Gamma \vdash u : A}{\Gamma \vdash t\,u : B}\ (\to_E)$$

Normalization

If $\Gamma \vdash t : A$, then t normalizes.

A somewhat obvious observation

Deduction rules

$$\frac{A \in \Gamma}{\Gamma \vdash A}\ (\mathrm{Ax})$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B}\ (\to_I)$$

$$\frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B}\ (\to_E)$$

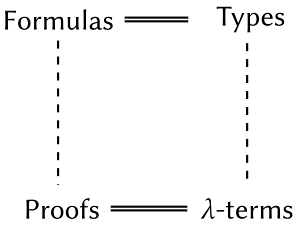
Typing rules

$$\frac{(x : A) \in \Gamma}{\Gamma \vdash x : A}\ (\mathrm{Ax})$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \to B}\ (\to_I)$$

$$\frac{\Gamma \vdash t : A \to B \quad \Gamma \vdash u : A}{\Gamma \vdash t\, u : B}\ (\to_E)$$

Proofs-as-programs



Proofs-as-programs

The Curry-Howard correspondence

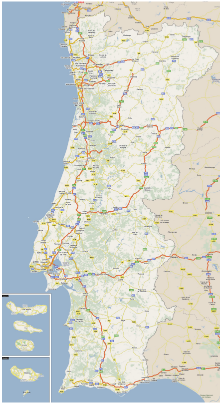
Mathematics	**Computer Science**
Proofs	Programs
Propositions	Types
Deduction rules	Typing rules
$\dfrac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B}\ (\Rightarrow_E)$	$\dfrac{\Gamma \vdash t : A \longrightarrow B \quad \Gamma \vdash u : A}{\Gamma \vdash t\, u : B}\ (\rightarrow_E)$
A implies B	function $A \longrightarrow B$
A and B	pair of A and B
$\forall x \in A. B(x)$	dependent product $\Pi x : A.B$

Benefits:

Program your proofs!	*Prove your programs!*

Typing vs. realizability

This is highly **syntactic**.
(provability)



Typing
$\vdash t : A$

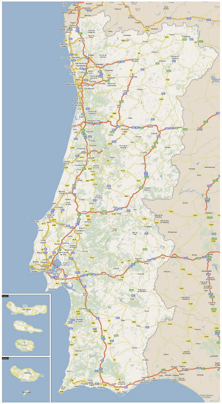
What about **semantics** ?
(validity)

Realizability
$t \Vdash A$

Typing vs. realizability

This is highly **syntactic**.
(provability)

What about **semantics** ?
(validity)



Typing

$\vdash t : A$

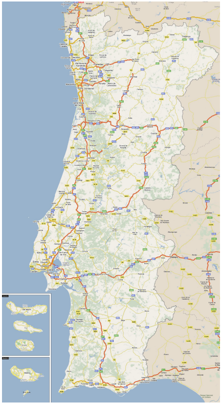
Realizability

$t \Vdash A$

Typing vs. realizability

This is highly **syntactic**.
(provability)

What about **semantics** ?
(validity)



$\subseteq$

Typing

$\vdash t : A$

Realizability

$t \Vdash A$

Realizability models

Krivine realizability:

$$A \mapsto \{t : t \Vdash A\}$$

(intuition: programs that share a common computational behavior given by A)

Tarski

$$A \mapsto |A| \in \mathbb{B}$$

(intuition: level of truthness)

Great news

Krivine realizability semantics gives surprisingly new models!

(in particular, provides us with a direct construction of $\mathcal{M} \vDash ZF_\varepsilon + \neg CH + \neg AC$)

Krivine realizability, a 3-steps recipe

1. an operational semantics
2. a logical language
3. formulas interpretation

Krivine realizability, a 3-steps recipe

1. an operational semantics (*a.k.a. the abstract Krivine machine*)

$$\begin{array}{lrcl} \text{PUSH}: & t\,u \star \pi & \to & t \star u \cdot \pi \\ \text{GRAB}: & \lambda x.t \star u \cdot \pi & \to & t\{x := u\} \star \pi \\ \text{SAVE}: & \mathbf{cc} \star t \cdot \pi & \to & t \star \mathbf{k}_\pi \cdot \pi \\ \text{RESTORE}: & \mathbf{k}_\pi \star t \cdot \rho & \to & t \star \pi \end{array}$$

2. a logical language (*a.k.a. a type system*)

1st-order terms	e	$::=$	$x \mid f(e_1, \dots, e_k)$
Formulas	A, B	$::=$	$X(e_1, \dots, e_k) \mid A \Rightarrow B \mid \forall x.A \mid \forall X.A$

3. formulas interpretation

Krivine realizability, a 3-steps recipe

1. an operational semantics (*a.k.a. the abstract Krivine machine*)

$$\begin{array}{lrcl} \textsc{Push}: & t\,u \star \pi & \rightarrow & t \star u \cdot \pi \\ \textsc{Grab}: & \lambda x.t \star u \cdot \pi & \rightarrow & t\{x := u\} \star \pi \\ \textsc{Save}: & \mathbf{cc} \star t \cdot \pi & \rightarrow & t \star \mathbf{k}_\pi \cdot \pi \\ \textsc{Restore}: & \mathbf{k}_\pi \star t \cdot \rho & \rightarrow & t \star \pi \end{array}$$

2. a logical language (*a.k.a. a type system*)
3. formulas interpretation

- falsity value $\|A\|$: stacks, opponent to A
- truth value $|A|$: terms, player of A
- pole $\perp\!\!\!\perp$: processes, referee

Krivine realizability, a 3-steps recipe

1. an operational semantics (*a.k.a. the abstract Krivine machine*)

PUSH	:	$t\,u \star \pi$	$\longrightarrow$	$t \star u \cdot \pi$
GRAB	:	$\lambda x.t \star u \cdot \pi$	$\longrightarrow$	$t\{x := u\} \star \pi$
SAVE	:	$\mathbf{cc} \star t \cdot \pi$	$\longrightarrow$	$t \star \mathbf{k}_\pi \cdot \pi$
RESTORE	:	$\mathbf{k}_\pi \star t \cdot \rho$	$\longrightarrow$	$t \star \pi$

2. a logical language (*a.k.a. a type system*)
3. formulas interpretation

- falsity value $\|A\|$: stacks, opponent to A
  - $\|A \Rightarrow B\| = \{t \cdot \pi : t \in |A| \wedge \pi \in \|B\|\}$
  - $\|\forall x.A\| = \bigcup_{n\in\mathbb{N}} \|A\{x := n\}\|$
- truth value $|A|$: $|A| = \|A\|^{\perp\!\!\!\perp} = \{t \in \Lambda : \forall \pi \in \|A\|, t \star \pi \in \perp\!\!\!\perp\}$
- pole $\perp\!\!\!\perp$: processes, referee

Krivine realizability, a 3-steps recipe

1. an operational semantics (*a.k.a. the abstract Krivine machine*)

$$\begin{array}{lrcl} \text{PUSH}: & t\,u \star \pi & \to & t \star u \cdot \pi \\ \text{GRAB}: & \lambda x.t \star u \cdot \pi & \to & t\{x := u\} \star \pi \\ \text{SAVE}: & \mathbf{cc} \star t \cdot \pi & \to & t \star \mathbf{k}_\pi \cdot \pi \\ \text{RESTORE}: & \mathbf{k}_\pi \star t \cdot \rho & \to & t \star \pi \end{array}$$

2. a logical language (*a.k.a. a type system*)
3. formulas interpretation

- falsity value $\|A\|$: stacks, opponent to A
- truth value $|A|$: $|A| = \|A\|^{\perp\!\!\!\perp} = \{t \in \Lambda : \forall \pi \in \|A\|, t \star \pi \in \perp\!\!\!\perp\}$
- pole $\perp\!\!\!\perp$: processes, referee

Adequacy

Typed terms are realizers.

Realizability models

Given the previous ingredients:

1. a calculus
2. its type system
3. an adequate interpretation of formula

one defines a model $\mathcal{M}_{\perp\!\!\perp}$ by:

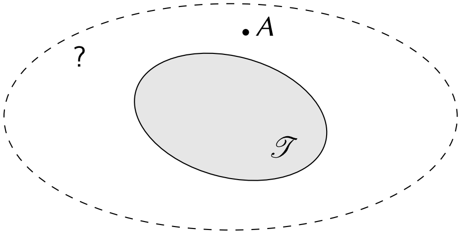
Realizability model

$$\mathcal{M}_{\perp\!\!\perp} \vDash A \quad \textit{iff} \quad \exists t \in |A|$$

i.e. A is realized

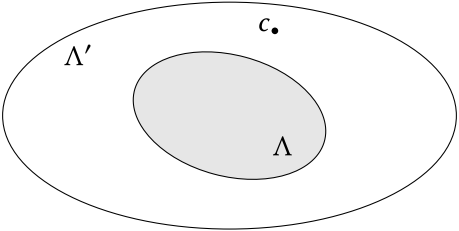
Product of realizability models

Extending Curry-Howard



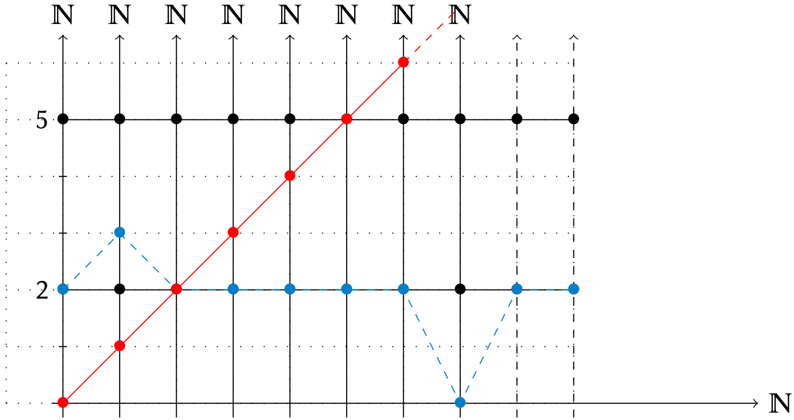
New axiom $\sim$ Programing primitive

Extending Curry-Howard

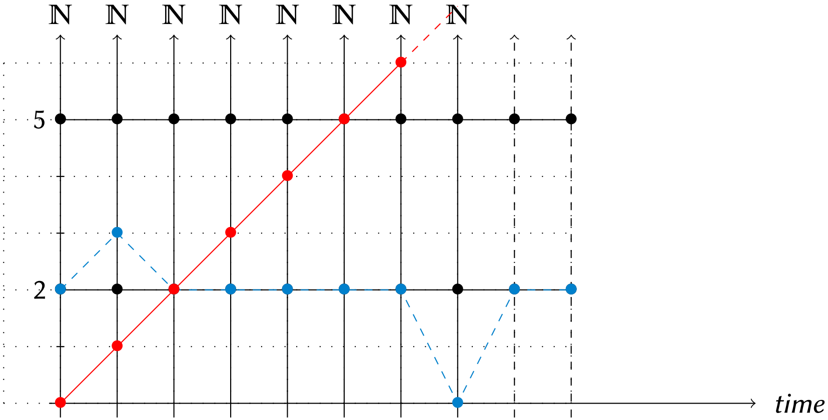


New axiom $\sim$ Programing primitive

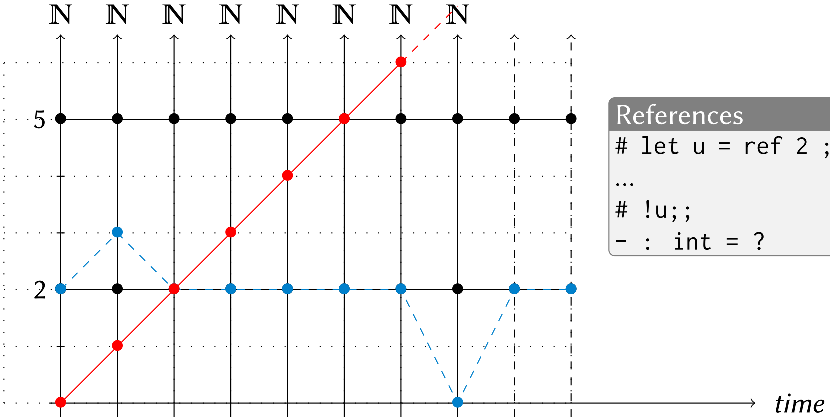
Intuition



Intuition



Intuition



References

```
# let u = ref 2 ;;
...
# !u;;
- : int = ?
```

Let's try!

Calculus: λ_c-calculus + states

$\mathcal{S}$: poset of states (in practice, $\mathbb{N}$)

PUSH :	$tu \star \pi \star s$	$\rightarrow$	$t \star u \cdot \pi \star s$
GRAB :	$\lambda x.t \star u \cdot \pi \star s$	$\rightarrow$	$t\{x := u\} \star \pi \star s$
SAVE :	$\mathbf{cc} \star t \cdot \pi \star s$	$\rightarrow$	$t \star \mathbf{k}_\pi \cdot \pi \star s$
RESTORE :	$\mathbf{k}_\pi \star t \cdot \rho \star s$	$\rightarrow$	$t \star \pi \star s$
GET :	$\mathbf{get} \star u \cdot \pi \star s$	$\rightarrow$	$u \star \overline{s} \cdot \pi \star s$
SET :	$\mathbf{set} \star \overline{s} \cdot u \cdot \pi \star s'$	$\rightarrow$	$u \star \pi \star \max(s, s')$

Semantics

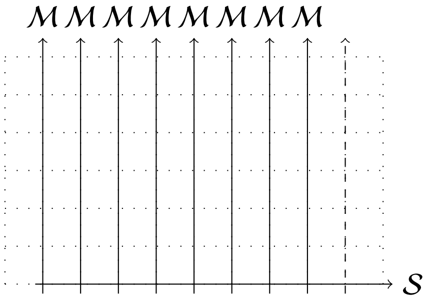
We want a *product*

Falsity values: $\mathcal{P}(\Pi)^{\mathcal{S}}$

Truth values: $\mathcal{P}(\Lambda)^{\mathcal{S}}$

Poles: glueing of $(\perp\!\!\!\perp_s)_{s\in\mathcal{S}}$?

Orthogonality: $(t,s)\perp\!\!\!\perp(\pi,s') := (s = s' \Rightarrow t \star \pi \star s \in \perp\!\!\!\perp)$

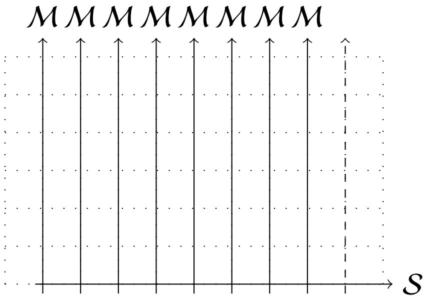


Individuals : $(x_s)_{s\in\mathcal{S}} \quad \equiv \quad x : \mathcal{S} \to \mathbb{N}$

Semantics

We want a *product*

Falsity values:	$\mathcal{P}(\Pi)^{\mathcal{S}} \equiv \mathcal{P}(\Pi \times \mathcal{S})$
Truth values:	$\mathcal{P}(\Lambda)^{\mathcal{S}} \equiv \mathcal{P}(\Lambda \times \mathcal{S})$
Poles:	glueing of $⫫ = (⫫_s)_{s \in \mathcal{S}}$
Orthogonality:	$(t, s) ⫫ (\pi, s') := (s = s' \Rightarrow t \star \pi \star s \in ⫫)$



Individuals : $(x_s)_{s \in \mathcal{S}} \equiv x : \mathcal{S} \to \mathbb{N}$

Semantics

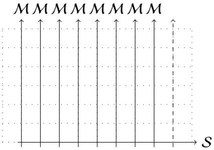
We want a *product*

Falsity values: $\mathcal{P}(\Pi)^{\mathcal{S}} \equiv \mathcal{P}(\Pi \times \mathcal{S})$

Truth values: $\mathcal{P}(\Lambda)^{\mathcal{S}} \equiv \mathcal{P}(\Lambda \times \mathcal{S})$

Poles: glueing of $⫫ = (⫫_s)_{s \in \mathcal{S}}$

Orthogonality: $(t, s) ⫫ (\pi, s') := (s = s' \Rightarrow t \star \pi \star s \in ⫫)$



Individuals : $(x_s)_{s \in \mathcal{S}} \equiv x : \mathcal{S} \to \mathbb{N}$

Predicates : $(X_s)_{s \in \mathcal{S}} \equiv X : \mathbb{N}^k \to \mathcal{P}(\Pi \times \mathcal{S})$

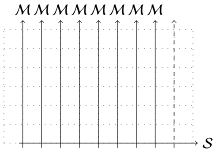
Glueing

$$X@(x_1, \ldots, x_k) = \{(\pi, s) : \pi \in X^s(x_1(s), \ldots, x_k(s))\}$$

Semantics

Individuals : $(x_s)_{s \in \mathcal{S}} \;\equiv\; x : \mathcal{S} \to \mathbb{N}$

Predicates : $(X_s)_{s \in \mathcal{S}} \;\equiv\; X : \mathbb{N}^k \to \mathcal{P}(\Pi \times \mathcal{S})$



Glueing

$$X@(x_1, \ldots, x_k) = \{(\pi, s) : \pi \in X^s(x_1(s), \ldots, x_k(s))\}$$

Lemma

A **internal**:

1. $(\pi, s) \in \|A\| \Leftrightarrow \pi \in \|A^s\|_s$
2. $(t, s) \in |A| \Leftrightarrow t \in |A^s|_s$

↬ **transfer** holds:

$$\forall^{\mathrm{st}} x. A(x) \Rightarrow \forall x. A(x) \qquad (A \text{ internal})$$

Soundness

Adequacy lemma

Typing rules are adequate except second-order elimination.

Explanation:

$$\frac{\Gamma \vdash t : \forall X.A}{\Gamma \vdash t : A\{X(x_1, \dots, x_k) := B\}}$$

$$X : \mathbb{N}^k \longrightarrow \mathcal{P}(\Pi) \text{ while } B : (\mathbb{N}^S)^k \longrightarrow \mathcal{P}(\Pi)$$

Valid if:

- B internal (glueing)
- B proposition ($k = 0$)

Soundness

Adequacy lemma

Typing rules are adequate except second-order elimination.

Explanation:

$$\frac{\Gamma \vdash t : \forall X.A}{\Gamma \vdash t : A\{X(x_1, \ldots, x_k) := B\}}$$

$$X : \mathbb{N}^k \to \mathcal{P}(\Pi) \text{ while } B : (\mathbb{N}^S)^k \to \mathcal{P}(\Pi)$$

Valid if:

- B internal (glueing)
- B proposition ($k = 0$)

Soundness

Adequacy lemma

Typing rules are adequate except second-order elimination.

Explanation:

$$\frac{\Gamma \vdash t : \forall X.A}{\Gamma \vdash t : A\{X(x_1, \ldots, x_k) := B\}}$$

$$X : \mathbb{N}^k \to \mathcal{P}(\Pi) \text{ while } B : (\mathbb{N}^S)^k \to \mathcal{P}(\Pi)$$

Valid if:

- B internal (glueing)
- B proposition ($k = 0$)

Natural numbers

We can define the predicate Nat(x), and show that:

Closure

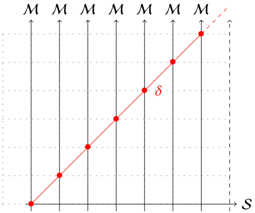
A internal, $s \in \mathcal{S}$:

1. $(\lambda x.(x\bar{0}), s) \Vdash \mathrm{Nat}(0)$
2. $(\mathbf{succ}, s) \Vdash \forall x.(\mathrm{Nat}(x) \Rightarrow \mathrm{Nat}(s(x)))$
3. $(\mathbf{rec}, s) \Vdash A(0) \Rightarrow \forall^{\mathrm{Nat}} y.(A(y) \Rightarrow A(s(y)) \Rightarrow \forall^{\mathrm{Nat}} x.A(x)$

Diagonal

$(\mathbf{get}, s) \Vdash \mathrm{Nat}(\delta)$

where $\delta : s \mapsto s$

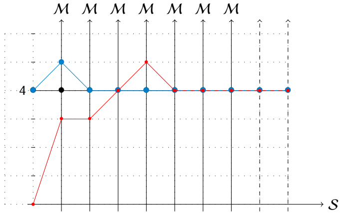


Looking for a quotient

(work in progress)

Quotienting

$$f \sim g := \exists s_0 . \forall s \geq s_0 . f(s) = g(s)$$



Definitions

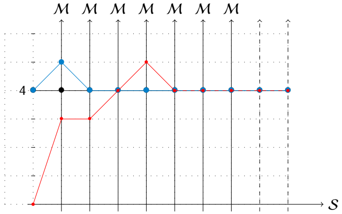
$$\begin{aligned} \|A\|^* &= \{\pi : \exists s_\pi, \forall s \geq s_\pi, (\pi, s) \in \|A\|\} \\ |A|^* &= \{t : \exists s_t, \forall s \geq s_t, (t, s) \in |A|\} \\ t \perp\!\!\!\perp^* \pi &= \exists s_t, \forall s \geq s_t, t \star \pi \star s \in \perp\!\!\!\perp \end{aligned}$$

Properties

1. $|\forall x A|^* \subsetneq \bigcap_{f \in \mathbb{N}^S} |A\{x := f\}|^*$
2. $t \in |A|^* \overset{\not\Leftarrow}{\Longrightarrow} t \in (\|A\|^*)^{\perp\!\!\!\perp^*}$

Quotienting

$$f \sim g := \exists s_0 . \forall s \geq s_0 . f(s) = g(s)$$



Definitions

$$\begin{aligned} \|A\|^* &= \{\pi : \exists s_\pi, \forall s \geq s_\pi, (\pi, s) \in \|A\|\} \\ |A|^* &= \{t : \exists s_t, \forall s \geq s_t, (t, s) \in |A|\} \\ t \perp\!\!\!\perp^* \pi &= \exists s_t, \forall s \geq s_t, t \star \pi \star s \in \perp\!\!\!\perp \end{aligned}$$

Properties

1. $|\forall x A|^* \subsetneq \bigcap_{f \in \mathbb{N}^{\mathcal{S}}} |A\{x := f\}|^*$
2. $t \in |A|^* \overset{\not\Leftarrow}{\Rightarrow} t \in (\|A\|^*)^{\perp\!\!\!\perp^*}$

Results

Adequacy

If A is internal and $\vdash t : A$, then $t \Vdash^* A$

Closure

A internal:

1. $\lambda x.(x\overline{0}) \Vdash^* \mathrm{Nat}(0)$
2. $\mathbf{succ} \Vdash^* \forall x.(\mathrm{Nat}(x) \Rightarrow \mathrm{Nat}(s(x)))$
3. $\mathbf{rec} \Vdash^* A(0) \Rightarrow \forall^{\mathrm{Nat}} y.(A(y) \Rightarrow A(s(y)) \Rightarrow \forall^{\mathrm{Nat}} x.A(x)$

Transfer

1. $|\forall x A(x)|^* = |\forall^{st} x A(x)|^*$
2. $\|\forall^{st} x A(x)\|^* = \|\forall x A(x)\|^*$

Standardization

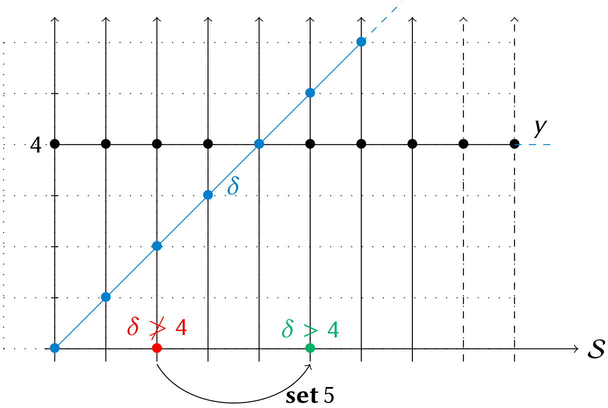
$$\exists^{st} A.\forall^{st} z.(A(z) \Leftrightarrow C(z)) \qquad \text{(for any } C\text{)}$$

Idealization

Diagonal

$\lambda n.(\textbf{succ}\ n\ \textbf{set}) \Vdash^* \forall^{\{\overline{\text{st}}\}} y.\delta > y \qquad \delta : s \mapsto s$

Proof:



Future work

- **Idealization**: for any internal relation?

$$\forall^{\mathrm{st}}(n \in \mathbb{N}).\exists^{\mathrm{st}}x.\forall^{\mathrm{st}}y.(y \leq n \Rightarrow R(x,y)) \Rightarrow \exists x.\forall^{\mathrm{st}}y.R(x,y)$$

- **Quotient**: unsatisfactory as such...

- **Next steps**:
  - consider the intuitionistic case
  - compare with existing related works
    (functional interp. / bounded realizability / ...)

- **Logical translation?**

New axiom $\rightsquigarrow$ Programming primitive

$\Updownarrow$ $\Updownarrow$

Logical translation? $\rightsquigarrow$ Program translation

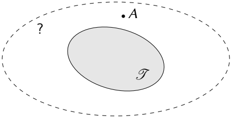
Future work

- **Idealization**: for any internal relation?

$$\forall^{st}(n \in \mathbb{N}).\exists^{st}x.\forall^{st}y.(y \leq n \Rightarrow R(x,y)) \Rightarrow \exists x.\forall^{st}y.R(x,y)$$

- **Quotient**: unsatisfactory as such...
- **Next steps**:
  - consider the intuitionistic case
  - compare with existing related works
    (functional interp. / bounded realizability / ...)
- **Logical translation?**

New axiom	~	Programing primitive
⇕		⇕
Logical translation?	~	Program translation



Future work

- **Idealization**: for any internal relation?

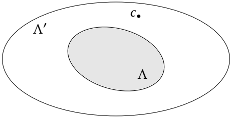
$$\forall^{st}(n \in \mathbb{N}).\exists^{st}x.\forall^{st}y.(y \leq n \Rightarrow R(x,y)) \Rightarrow \exists x.\forall^{st}y.R(x,y)$$

- **Quotient**: unsatisfactory as such...
- **Next steps**:
  - consider the intuitionistic case
  - compare with existing related works
    (functional interp. / bounded realizability / ...)
- **Logical translation?**

New axiom ~ Programing primitive

$\Updownarrow$ $\Updownarrow$

Logical translation? ~ Program translation



Future work

- **Idealization**: for any internal relation?

$$\forall^{st}(n \in \mathbb{N}).\exists^{st}x.\forall^{st}y.(y \leq n \Rightarrow R(x,y)) \Rightarrow \exists x.\forall^{st}y.R(x,y)$$

- **Quotient**: unsatisfactory as such...

- **Next steps**:
  - consider the intuitionistic case
  - compare with existing related works

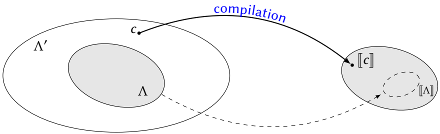
    (functional interp. / bounded realizability / ...)

- **Logical translation?**

New axiom ~ Programing primitive

⇕ ⇕

Logical translation? ~ Program translation



Future work

- **Idealization**: for any internal relation?

$$\forall^{st}(n \in \mathbb{N}).\exists^{st} x.\forall^{st} y.(y \leq n \Rightarrow R(x,y)) \Rightarrow \exists x.\forall^{st} y.R(x,y)$$

- **Quotient**: unsatisfactory as such...
- **Next steps**:
  - consider the intuitionistic case
  - compare with existing related works
    (functional interp. / bounded realizability / ...)
- **Logical translation?**

New axiom	$\sim$	Programing primitive
$\Updownarrow$		$\Updownarrow$
Logical translation?	$\sim$	Program translation