Mechanizing Second-Order Logic in Coq

Final Bachelor Talk

Mark Koch
Advisor: Dominik Kirst
Supervisor: Gert Smolka

August 26, 2021

Saarland University, Programming Systems Lab

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

$$\forall P.\, P(0) \to (\forall x.\, P(x) \to P(x+1)) \to \forall x.\; P(x)$$

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

Second-order logic

Quantification over individuals & their properties:

$$\forall P.\, P(0) \to (\forall x.\, P(x) \to P(x+1)) \to \forall x.\ P(x)$$

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

Second-order logic

Quantification over individuals & their properties:

$$\forall P.\, P(0) \to (\forall x.\, P(x) \to P(x+1)) \to \forall x.\ P(x)$$

$\Rightarrow$ Drastically alters meta-behaviour!

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

Second-order logic

Quantification over individuals & their properties:

$$\forall P.\, P(0) \to (\forall x.\, P(x) \to P(x+1)) \to \forall x.\ P(x)$$

$\Rightarrow$ Drastically alters meta-behaviour!

Our goals:

- Formalize meta-theoretical properties of SOL in Coq's constructive type theory and compare to FOL

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

Second-order logic

Quantification over individuals & their properties:

$$\forall P.\, P(0) \to (\forall x.\, P(x) \to P(x+1)) \to \forall x.\ P(x)$$

$\Rightarrow$ Drastically alters meta-behaviour!

Our goals:

- Formalize meta-theoretical properties of SOL in Coq's constructive type theory and compare to FOL
- Show that SOL with Henkin semantics reduces to FOL

First-order logic

Quantification over individuals:

$$\forall x.\, \exists y.\, y > x$$

Second-order logic

Quantification over individuals & their properties:

$$\forall P.\, P(0) \to (\forall x.\, P(x) \to P(x+1)) \to \forall x.\ P(x)$$

$\Rightarrow$ Drastically alters meta-behaviour!

Our goals:

- Formalize meta-theoretical properties of SOL in Coq's constructive type theory and compare to FOL
- Show that SOL with Henkin semantics reduces to FOL
- Mechanize undecidability results for SOL

Related work

- We largely follow [Shapiro, 1991] and [Nour and Raffalli, 2003]

Related work

- We largely follow [Shapiro, 1991] and [Nour and Raffalli, 2003]
- Mechanization based on FOL development in [Kirst and Larchey-Wendling, 2020] among others

Related work

- We largely follow [Shapiro, 1991] and [Nour and Raffalli, 2003]
- Mechanization based on FOL development in [Kirst and Larchey-Wendling, 2020] among others
- Synthetic undecidability and incompleteness of axiom systems by [Kirst and Hermes, 2021]

Related work

- We largely follow [Shapiro, 1991] and [Nour and Raffalli, 2003]
- Mechanization based on FOL development in [Kirst and Larchey-Wendling, 2020] among others
- Synthetic undecidability and incompleteness of axiom systems by [Kirst and Hermes, 2021]
- Synthetic computability theory [Bauer, 2006, Forster et al., 2019a] as employed in the Coq Library of Undecidability Proofs [Forster et al., 2019b].

Definition (Syntax)

$$t ::= x_i \mid \mathcal{F}\ \boldsymbol{t} \qquad (\mathcal{F} : \Sigma_f)$$

$$\varphi, \psi ::= \dot{\bot} \mid \mathcal{P}\ \boldsymbol{t} \mid p_i^n\ \boldsymbol{t} \mid \varphi \mathrel{\dot{\to}} \psi \mid \varphi\ \dot{\wedge}\ \psi \mid \varphi\ \dot{\vee}\ \psi \qquad (\mathcal{P} : \Sigma_p)$$

$$\mid \dot{\forall}\ \varphi \mid \dot{\exists}\ \varphi \mid \dot{\forall}_p^n\ \varphi \mid \dot{\exists}_p^n\ \varphi \qquad (i, n : \mathbb{N})$$

Standard Tarski Semantics

Definition (Syntax)

$$t ::= x_i \mid \mathcal{F}\ \boldsymbol{t} \qquad (\mathcal{F} : \Sigma_f)$$

$$\varphi, \psi ::= \dot{\bot} \mid \mathcal{P}\ \boldsymbol{t} \mid p_i^n\ \boldsymbol{t} \mid \varphi \mathrel{\dot{\to}} \psi \mid \varphi\ \dot{\wedge}\ \psi \mid \varphi\ \dot{\vee}\ \psi \qquad (\mathcal{P} : \Sigma_p)$$

$$\mid \dot{\forall}\ \varphi \mid \dot{\exists}\ \varphi \mid \dot{\forall}_p^n\ \varphi \mid \dot{\exists}_p^n\ \varphi \qquad (i, n : \mathbb{N})$$

Definition (Standard Tarski Semantics)

A model $\mathcal{M}$ consists of a domain D and interpretation $\mathcal{I}$ for function and predicate symbols.

Definition (Syntax)

$$t ::= x_i \mid \mathcal{F}\ \boldsymbol{t} \qquad (\mathcal{F} : \Sigma_f)$$

$$\varphi, \psi ::= \dot{\bot} \mid \mathcal{P}\ \boldsymbol{t} \mid p_i^n\ \boldsymbol{t} \mid \varphi \mathrel{\dot{\to}} \psi \mid \varphi\ \dot{\wedge}\ \psi \mid \varphi\ \dot{\vee}\ \psi \qquad (\mathcal{P} : \Sigma_p)$$

$$\mid \dot{\forall}\ \varphi \mid \dot{\exists}\ \varphi \mid \dot{\forall}_p^n\ \varphi \mid \dot{\exists}_p^n\ \varphi \qquad (i, n : \mathbb{N})$$

Definition (Standard Tarski Semantics)

A model $\mathcal{M}$ consists of a domain D and interpretation $\mathcal{I}$ for function and predicate symbols. Predicate quantifiers range over all properties on D:

$$\rho \vDash \dot{\forall}_p^n \varphi \ :=\ \forall P^{D^n \to \mathsf{Prop}}.\ P \cdot \rho \vDash \varphi$$

Second-order Peano Arithmetic

Standard semantics can uniquely characterize the numbers via PA_2:

Zero Addition : $\dot{\forall} x.\, 0 + x \equiv x$

Addition Recursion : $\dot{\forall} xy.\, (Sx) + y \equiv S(x + y)$

Disjointness : $\dot{\forall} x.\, 0 \equiv Sx \mathrel{\dot{\to}} \dot{\bot}$

Equlity Reflexive : $\dot{\forall} x.\, x \equiv x$

Zero Multiplication : $\dot{\forall} x.\, 0 \cdot x \equiv 0$

Multiplication Recursion : $\dot{\forall} xy.\, (Sx) \cdot y \equiv y + x \cdot y$

Successor Injective : $\dot{\forall} xy.\, Sx \equiv Sy \mathrel{\dot{\to}} x \equiv y$

Equlity Symmetric : $\dot{\forall} xy.\, x \equiv y \mathrel{\dot{\to}} y \equiv x$

Induction : $\dot{\forall} P.\, P(0) \mathrel{\dot{\to}} (\dot{\forall} x.\, P(x) \mathrel{\dot{\to}} P(Sx)) \mathrel{\dot{\to}} \dot{\forall} x.\, P(x)$

Second-order Peano Arithmetic

Standard semantics can uniquely characterize the numbers via PA_2:

Zero Addition : $\dot{\forall} x.\, 0 + x \equiv x$

Addition Recursion : $\dot{\forall} xy.\, (Sx) + y \equiv S(x + y)$

Disjointness : $\dot{\forall} x.\, 0 \equiv Sx \mathrel{\dot{\to}} \dot{\bot}$

Equlity Reflexive : $\dot{\forall} x.\, x \equiv x$

Zero Multiplication : $\dot{\forall} x.\, 0 \cdot x \equiv 0$

Multiplication Recursion : $\dot{\forall} xy.\, (Sx) \cdot y \equiv y + x \cdot y$

Successor Injective : $\dot{\forall} xy.\, Sx \equiv Sy \mathrel{\dot{\to}} x \equiv y$

Equlity Symmetric : $\dot{\forall} xy.\, x \equiv y \mathrel{\dot{\to}} y \equiv x$

Induction : $\dot{\forall} P.\, P(0) \mathrel{\dot{\to}} (\dot{\forall} x.\, P(x) \mathrel{\dot{\to}} P(Sx)) \mathrel{\dot{\to}} \dot{\forall} x.\, P(x)$

Theorem (Categoricity)

All models of PA_2 are isomorphic. We say that PA_2 is **categorical**.

Löwenheim-Skolem

Not possible for FOL because of **upward Löwenheim-Skolem theorem**:

"*Every first-order theory with an infinite model has models of every greater infinite cardinality.*"

Löwenheim-Skolem

Not possible for FOL because of **upward Löwenheim-Skolem theorem**:

"*Every first-order theory with an infinite model has models of every greater infinite cardinality.*"

Theorem (Failure of SOL Upward Löwenheim-Skolem)

SOL with standard semantics does not have the upward Löwenheim-Skolem property.

Löwenheim-Skolem

Not possible for FOL because of **upward Löwenheim-Skolem theorem**:

> "*Every first-order theory with an infinite model has models of every greater infinite cardinality.*"

Theorem (Failure of SOL Upward Löwenheim-Skolem)

SOL with standard semantics does not have the upward Löwenheim-Skolem property.

Failure of downwards direction via categoricity of second-order real analysis [Shapiro, 1991] or set theory [Kirst and Smolka, 2017]

Failure of Compactness

FOL is **compact**:

> "$\mathcal{T}$ *has a model if every finite subset of* $\mathcal{T}$ *has a model.*"

Failure of Compactness

FOL is **compact**:

"*$\mathcal{T}$ has a model if every finite subset of $\mathcal{T}$ has a model.*"

Theorem (Failure of Compactness).

SOL is not compact for standard semantics.

Failure of Compactness

FOL is **compact**:

"$\mathcal{T}$ *has a model if every finite subset of* $\mathcal{T}$ *has a model.*"

Theorem (Failure of Compactness).

SOL is not compact for standard semantics.

Proof.

Consider the theory $\mathcal{T}_{\neq} := \mathrm{PA}_2, x_0 \neq O, x_0 \neq S\,O, x_0 \neq S\,(S\,O), \ldots$

Failure of Compactness

FOL is **compact**:

"$\mathcal{T}$ has a model if every finite subset of $\mathcal{T}$ has a model."

Theorem (Failure of Compactness).

SOL is not compact for standard semantics.

Proof.

Consider the theory $\mathcal{T}_{\neq} := \mathsf{PA}_2, x_0 \neq O, x_0 \neq S\,O, x_0 \neq S\,(S\,O), \ldots$

- Every finite subset of $\mathcal{T}_{\neq}$ has a model.

Failure of Compactness

FOL is **compact**:

"$\mathcal{T}$ *has a model if every finite subset of* $\mathcal{T}$ *has a model.*"

Theorem (Failure of Compactness).

SOL is not compact for standard semantics.

Proof.

Consider the theory $\mathcal{T}_{\neq} := \mathsf{PA}_2, x_0 \neq O, x_0 \neq S\,O, x_0 \neq S\,(S\,O), \ldots$

- Every finite subset of $\mathcal{T}_{\neq}$ has a model.
- $\mathcal{T}_{\neq}$ itself does not have a model.

Failure of Compactness

FOL is **compact**:

"$\mathcal{T}$ *has a model if every finite subset of* $\mathcal{T}$ *has a model.*"

Theorem (Failure of Compactness).

SOL is not compact for standard semantics.

Proof.

Consider the theory $\mathcal{T}_{\neq} := \mathsf{PA}_2, x_0 \neq O, x_0 \neq S\,O, x_0 \neq S\,(S\,O), \ldots$

- Every finite subset of $\mathcal{T}_{\neq}$ has a model.
- $\mathcal{T}_{\neq}$ itself does not have a model. Otherwise, $\mathbb{N}$ would also need to be a model because of categoricity which is not possible. □

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete.

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete.

- Finitary Completeness: $A \vDash \varphi \to A \vdash \varphi$

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

- Finitary Completeness: $A \vDash \varphi \to A \vdash \varphi$
- Lift $\vdash$ to theories: $\mathcal{T} \vdash \varphi := \exists A.\, A \subseteq_{\mathrm{fin}} \mathcal{T} \wedge A \vdash \varphi$

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

- Finitary Completeness: $A \vDash \varphi \to A \vdash \varphi$
- Lift $\vdash$ to theories: $\mathcal{T} \vdash \varphi := \exists A.\, A \subseteq_{\mathrm{fin}} \mathcal{T} \wedge A \vdash \varphi$
- Infinitary Completeness: $\mathcal{T} \vDash \varphi \to \mathcal{T} \vdash \varphi$

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$.

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \models \dot{\bot}$$

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \vDash \dot{\bot} \xrightarrow{\text{Complete}} \mathcal{T}_{\neq} \vdash \dot{\bot}$$

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \vDash \dot{\bot} \xrightarrow{\text{Complete}} \mathcal{T}_{\neq} \vdash \dot{\bot} \xrightarrow{\text{Def}} \begin{array}{c} A \vdash \dot{\bot} \\ \text{for some } A \subseteq_{\text{fin}} \mathcal{T}_{\neq} \end{array}$$

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \vDash \dot{\bot} \xrightarrow{\text{Complete}} \mathcal{T}_{\neq} \vdash \dot{\bot} \xrightarrow{\text{Def}} \begin{array}{c} A \vdash \dot{\bot} \\ \text{for some } A \subseteq_{\text{fin}} \mathcal{T}_{\neq} \end{array} \xrightarrow{\text{Sound}} A \vDash \dot{\bot}$$

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \vDash \dot{\bot} \xrightarrow{\text{Complete}} \mathcal{T}_{\neq} \vdash \dot{\bot} \xrightarrow{\text{Def}} \begin{array}{c} A \vdash \dot{\bot} \\ \text{for some } A \subseteq_{\text{fin}} \mathcal{T}_{\neq} \end{array} \xrightarrow{\text{Sound}} A \vDash \dot{\bot}$$

But $A \subseteq_{\text{fin}} \mathcal{T}_{\neq}$ has a model. □

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \vDash \dot{\bot} \xrightarrow{\text{Complete}} \mathcal{T}_{\neq} \vdash \dot{\bot} \xrightarrow{\text{Def}} \begin{array}{c} A \vdash \dot{\bot} \\ \text{for some } A \subseteq_{\text{fin}} \mathcal{T}_{\neq} \end{array} \xrightarrow{\text{Sound}} A \vDash \dot{\bot}$$

But $A \subseteq_{\text{fin}} \mathcal{T}_{\neq}$ has a model. □

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\mathrm{form}) \to \mathrm{form} \to \mathrm{Prop}$ is not infinitary complete for decidable theories.

Proof.

Let $\vdash$ be sound and infinitary complete.

Previous proof: There is no model of $\mathcal{T}_{\neq}$. Thus

$$\mathcal{T}_{\neq} \vDash \dot{\bot} \xrightarrow{\text{Complete}} \mathcal{T}_{\neq} \vdash \dot{\bot} \xrightarrow{\text{Def}} \underset{\text{for some } A \subseteq_{\text{fin}} \mathcal{T}_{\neq}}{A \vdash \dot{\bot}} \xrightarrow{\text{Sound}} A \vDash \dot{\bot}$$

But $A \subseteq_{\text{fin}} \mathcal{T}_{\neq}$ has a model. □

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete for decidable theories.

- Remarkably simple and as far as we can tell not discussed in literature

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete for decidable theories.

- Remarkably simple and as far as we can tell not discussed in literature
- No computability requirements on $\vdash$!

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete for decidable theories.

- Remarkably simple and as far as we can tell not discussed in literature
- No computability requirements on $\vdash$!
- But does not rule out completeness for finite contexts:
  The "system" $A \vdash \varphi := A \vDash \varphi$ is sound and finitary complete

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete for decidable theories.

- Remarkably simple and as far as we can tell not discussed in literature
- No computability requirements on $\vdash$!
- But does not rule out completeness for finite contexts:
  The "system" $A \vdash \varphi := A \vDash \varphi$ is sound and finitary complete

  $\Rightarrow$ For finitary incompleteness we need to be stricter on what constitutes deduction:

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete for decidable theories.

- Remarkably simple and as far as we can tell not discussed in literature
- No computability requirements on $\vdash$!
- But does not rule out completeness for finite contexts:
  The "system" $A \vdash \varphi := A \vDash \varphi$ is sound and finitary complete

  $\Rightarrow$ For finitary incompleteness we need to be stricter on what constitutes deduction: Enumerability of $\vdash$ does the trick!

Infinitary Incompleteness

Theorem (Infinitary Incompleteness).

Every sound second-order deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ is not infinitary complete for decidable theories.

- Remarkably simple and as far as we can tell not discussed in literature
- No computability requirements on $\vdash$!
- But does not rule out completeness for finite contexts:
  The "system" $A \vdash \varphi := A \vDash \varphi$ is sound and finitary complete

  $\Rightarrow$ For finitary incompleteness we need to be stricter on what constitutes deduction: Enumerability of $\vdash$ does the trick!

  However requires much more involved proof!

Finitary Incompleteness

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable

Finitary Incompleteness

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Finitary Incompleteness

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Proof Sketch.

Via Reduction from Hilbert's tenth problem [Kirst and Hermes, 2021]

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Proof Sketch.

Via Reduction from Hilbert's tenth problem [Kirst and Hermes, 2021]

$$\underbrace{x+2}_{p} = \underbrace{y^2+z}_{q} \quad \rightsquigarrow \quad \varphi_{p,q} := \dot{\exists} xyz.\, x + S\,(S\,O) \equiv y \cdot y + z$$

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Proof Sketch.

Via Reduction from Hilbert's tenth problem [Kirst and Hermes, 2021]

$$\underbrace{x+2}_{p} = \underbrace{y^2+z}_{q} \quad \rightsquigarrow \quad \varphi_{p,q} := \dot{\exists} xyz.\, x + S\,(S\,O) \equiv y \cdot y + z$$

Enumerator for $\lambda\varphi.\, \mathbb{N} \vDash \varphi$ would yield decider for H_{10} via Post's theorem:

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Proof Sketch.

Via Reduction from Hilbert's tenth problem [Kirst and Hermes, 2021]

$$\underbrace{x+2}_{p} = \underbrace{y^2+z}_{q} \quad \rightsquigarrow \quad \varphi_{p,q} := \dot{\exists} xyz.\, x + S\,(S\,O) \equiv y \cdot y + z$$

Enumerator for $\lambda\varphi.\, \mathbb{N} \vDash \varphi$ would yield decider for H_{10} via Post's theorem:

- Enumerate H_{10} via $\mathbb{N} \vDash \varphi_{p,q}$.
- Enumerate $\overline{\mathrm{H}_{10}}$ via $\mathbb{N} \vDash \dot{\neg}\varphi_{p,q}$.

Finitary Incompleteness

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Theorem (Finitary Incompleteness)

Existence of a sound, enumerable and finitary complete deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ implies decidability of the halting problem under MP.

Lemma

The set of closed first-order statements that hold in $\mathbb{N}$ is not enumerable, in that enumerability implies decidability of the halting problem under MP.

Theorem (Finitary Incompleteness)

Existence of a sound, enumerable and finitary complete deduction system $\vdash : \mathcal{L}(\text{form}) \to \text{form} \to \text{Prop}$ implies decidability of the halting problem under MP.

Proof.

$\vdash$ yields enumerator for $\lambda\varphi.\text{PA}_2 \vDash \varphi$ and thus all truths in $\mathbb{N}$. $\square$

Undecidability

Theorem (Undecidability).

Validity and satisfiability in PA_2 and in the empty signature is undecidable.

Undecidability

Theorem (Undecidability).

Validity and satisfiability in PA_2 and in the empty signature is undecidable.

Proof Sketch.

- $p = q$ has a solution iff $\varphi_{p,q}$ is valid / satisfiable in PA_2.

Theorem (Undecidability).

Validity and satisfiability in PA_2 and in the empty signature is undecidable.

Proof Sketch.

- $p = q$ has a solution iff $\varphi_{p,q}$ is valid / satisfiable in PA_2.
- $p = q$ has a solution iff $\dot{\forall}\, f_0\, f_S\, f_+\, f_\times\, P_\equiv.\, \mathrm{PA}_2' \dot{\to} \varphi'_{p,q}$ is valid. □

Theorem (Undecidability).

Validity and satisfiability in PA_2 and in the empty signature is undecidable.

Proof Sketch.

- $p = q$ has a solution iff $\varphi_{p,q}$ is valid / satisfiable in PA_2.
- $p = q$ has a solution iff $\dot{\forall}\, f_0\, f_S\, f_+\, f_\times\, P_\equiv.\, \mathrm{PA}_2' \dot{\to} \varphi'_{p,q}$ is valid. □

Corollary (Non-Enumerability).

Those problems are also not enumerable under MP.

Henkin Semantics

Instead of quantifying over all predicates, specify a universe $\mathbb{U}$:

Instead of quantifying over all predicates, specify a universe $\mathbb{U}$:

Definition (Henkin Semantics).

A Henkin model $\mathcal{H}$ specifies a set of relations $\mathbb{U}_n : (D^n \to \text{Prop}) \to \text{Prop}$ that constrain the predicates that are quantified over, i.e

$$\rho \vDash \dot{\exists}^n_p \varphi \;:=\; \exists P^{D^n \to \text{Prop}}.\, \mathbb{U}_n\, P \wedge P \cdot \rho \vDash \varphi.$$

Instead of quantifying over all predicates, specify a universe $\mathbb{U}$:

Definition (Henkin Semantics).

A Henkin model $\mathcal{H}$ specifies a set of relations $\mathbb{U}_n : (D^n \to \text{Prop}) \to \text{Prop}$ that constrain the predicates that are quantified over, i.e

$$\rho \vDash \dot{\exists}^n_p \varphi \;:=\; \exists P^{D^n \to \text{Prop}}.\, \mathbb{U}_n\, P \wedge P \cdot \rho \vDash \varphi.$$

$\mathbb{U}_n$ should satisfy comprehension, i.e. it must at least contain all second-order definable properties.

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot{\forall} x.\, \dot{\exists} P.\, P(x, x) \quad \rightsquigarrow \quad \dot{\forall} x.\, \dot{\exists} p.\, \mathrm{App}_2(p, x, x)$$

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot{\forall} x.\, \dot{\exists} P.\, P(x, x) \quad \rightsquigarrow \quad \dot{\forall} x.\, \dot{\exists} p.\, \mathrm{App}_2(p, x, x)$$

x and p represent individuals and predicates at the same time.

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot\forall x.\, \dot\exists P.\, P(x,x) \quad \rightsquigarrow \quad \dot\forall x.\, \dot\exists p.\, \mathrm{App}_2(p,x,x)$$

x and p represent individuals and predicates at the same time.

$$\mathcal{T} \vDash_2 \varphi \iff (\mathcal{T} \cup \mathrm{Comprehension})^\star \vDash_1 \varphi^\star$$

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot\forall x.\, \dot\exists P.\, P(x,x) \quad \rightsquigarrow \quad \dot\forall x.\, \dot\exists p.\, \mathsf{App}_2(p,x,x)$$

x and p represent individuals and predicates at the same time.

$$\mathcal{T} \vDash_2 \varphi \iff (\mathcal{T} \cup \mathsf{Comprehension})^\star \vDash_1 \varphi^\star$$

$$\mathcal{T} \vdash_2 \varphi \impliedby (\mathcal{T} \cup \mathsf{Comprehension})^\star \vdash_1 \varphi^\star$$

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot\forall x.\, \dot\exists P.\, P(x,x) \quad \rightsquigarrow \quad \dot\forall x.\, \dot\exists p.\, \mathrm{App}_2(p,x,x)$$

x and p represent individuals and predicates at the same time.

$$\mathcal{T} \vDash_2 \varphi \iff (\mathcal{T} \cup \mathrm{Comprehension})^\star \vDash_1 \varphi^\star$$

FOL completeness (LEM)
[Forster et al., 2021]

$$\mathcal{T} \vdash_2 \varphi \longleftarrow (\mathcal{T} \cup \mathrm{Comprehension})^\star \vdash_1 \varphi^\star$$

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot{\forall} x.\, \dot{\exists} P.\, P(x,x) \quad \rightsquigarrow \quad \dot{\forall} x.\, \dot{\exists} p.\, \mathrm{App}_2(p,x,x)$$

x and p represent individuals and predicates at the same time.

$$\mathcal{T} \vDash_2 \varphi \iff (\mathcal{T} \cup \text{Comprehension})^\star \vDash_1 \varphi^\star$$

FOL completeness (LEM)
[Forster et al., 2021]

$$\mathcal{T} \vdash_2 \varphi \longleftarrow (\mathcal{T} \cup \text{Comprehension})^\star \vdash_1 \varphi^\star$$

Theorem (Completeness)

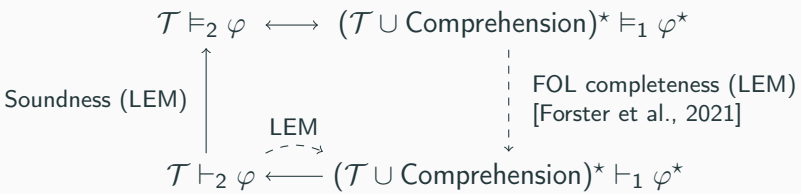
If FOL is complete, then so is SOL with Henkin semantics.

Reduction to FOL [Nour and Raffalli, 2003]

Turn φ into $\varphi^\star$ by replacing predicate quantifiers with individual ones:

$$\dot{\forall} x.\, \dot{\exists} P.\, P(x,x) \quad \rightsquigarrow \quad \dot{\forall} x.\, \dot{\exists} p.\, \mathrm{App}_2(p,x,x)$$

x and p represent individuals and predicates at the same time.



Theorem (Completeness)

If FOL is complete, then so is SOL with Henkin semantics.

Compactness

Theorem

If FOL is compact, then so is SOL with Henkin semantics.

Compactness

Theorem

If FOL is compact, then so is SOL with Henkin semantics.

Proof Sketch.

Assume every finite $A \subseteq \mathcal{T}$ has a Henkin model. We want to show

$$\mathcal{T} \text{ has Henkin model}$$

Compactness

Theorem

If FOL is compact, then so is SOL with Henkin semantics.

Proof Sketch.

Assume every finite $A \subseteq \mathcal{T}$ has a Henkin model. We want to show

$$\mathcal{T} \text{ has Henkin model}$$
$$\uparrow$$
$$(\mathcal{T} \cup \text{Comprehension})^{\star} \text{ has first-order model}$$

Compactness

Theorem

If FOL is compact, then so is SOL with Henkin semantics.

Proof Sketch.

Assume every finite $A \subseteq \mathcal{T}$ has a Henkin model. We want to show

$$\begin{array}{c} \mathcal{T} \text{ has Henkin model} \\ \uparrow \\ (\mathcal{T} \cup \text{Comprehension})^\star \text{ has first-order model} \\ \uparrow \\ A^\star_{\mathcal{T}} + A^\star_{\mathcal{C}} \subseteq (\mathcal{T} \cup \text{Comprehension})^\star \text{ has first-order model} \end{array}$$

Compactness

Theorem

If FOL is compact, then so is SOL with Henkin semantics.

Proof Sketch.

Assume every finite $A \subseteq \mathcal{T}$ has a Henkin model. We want to show

$$\mathcal{T} \text{ has Henkin model}$$
$$\uparrow$$
$$(\mathcal{T} \cup \text{Comprehension})^\star \text{ has first-order model}$$
$$\uparrow$$
$$A^\star_{\mathcal{T}} + A^\star_{\mathcal{C}} \subseteq (\mathcal{T} \cup \text{Comprehension})^\star \text{ has first-order model}$$
$$\uparrow$$
$$A_{\mathcal{T}} \text{ has Henkin model}$$

Löwenheim-Skolem

We combine upward and downward in one property: "*Every theory with an infinite model has models of every infinite cardinality.*"

Theorem (Löwenheim-Skolem)

If FOL has the Löwenheim-Skolem property, then so has SOL with Henkin semantics.

Löwenheim-Skolem

We combine upward and downward in one property: "*Every theory with an infinite model has models of every infinite cardinality.*"

Theorem (Löwenheim-Skolem)

If FOL has the Löwenheim-Skolem property, then so has SOL with Henkin semantics.

Proof.

Suppose $\mathcal{T}$ has an infinite Henkin model $\mathcal{H}$.

Löwenheim-Skolem

We combine upward and downward in one property: "*Every theory with an infinite model has models of every infinite cardinality.*"

Theorem (Löwenheim-Skolem)

If FOL has the Löwenheim-Skolem property, then so has SOL with Henkin semantics.

Proof.

Suppose $\mathcal{T}$ has an infinite Henkin model $\mathcal{H}$. Then $\mathcal{H}^\star$ is an infinite model of $\mathcal{T}^\star$

Löwenheim-Skolem

We combine upward and downward in one property: "*Every theory with an infinite model has models of every infinite cardinality.*"

Theorem (Löwenheim-Skolem)

If FOL has the Löwenheim-Skolem property, then so has SOL with Henkin semantics.

Proof.

Suppose $\mathcal{T}$ has an infinite Henkin model $\mathcal{H}$. Then $\mathcal{H}^\star$ is an infinite model of $\mathcal{T}^\star$ and $\mathcal{T}^\star$ has models of every infinite cardinality.

Löwenheim-Skolem

We combine upward and downward in one property: "*Every theory with an infinite model has models of every infinite cardinality.*"

Theorem (Löwenheim-Skolem)

If FOL has the Löwenheim-Skolem property, then so has SOL with Henkin semantics.

Proof.

Suppose $\mathcal{T}$ has an infinite Henkin model $\mathcal{H}$. Then $\mathcal{H}^\star$ is an infinite model of $\mathcal{T}^\star$ and $\mathcal{T}^\star$ has models of every infinite cardinality. Those can again be converted into Henkin models of $\mathcal{T}$. □

Löwenheim-Skolem

We combine upward and downward in one property: "*Every theory with an infinite model has models of every infinite cardinality.*"

Theorem (Löwenheim-Skolem)

If FOL has the Löwenheim-Skolem property, then so has SOL with Henkin semantics.

Proof.

Suppose $\mathcal{T}$ has an infinite Henkin model $\mathcal{H}$. Then $\mathcal{H}^\star$ is an infinite model of $\mathcal{T}^\star$ and $\mathcal{T}^\star$ has models of every infinite cardinality. Those can again be converted into Henkin models of $\mathcal{T}$. □

$\Rightarrow$ No categorical axiomatization of $\mathbb{N}$ possible!

Conclusion

Contributions: To the best of our knowledge, first mechanization of SOL.

Conclusion

Contributions: To the best of our knowledge, first mechanization of SOL.

- Formalized the following meta-theoretic properties:

Semantics	$\mathbb{N}$ Categorical	Completeness	Compactness	Löwenheim-Skolem
Standard	✓	✗	✗	✗ (upward)
Henkin	✗	✓	✓	(✓)

✓ = holds ✗ = does not hold

Conclusion

Contributions: To the best of our knowledge, first mechanization of SOL.

- Formalized the following meta-theoretic properties:

Semantics	$\mathbb{N}$ Categorical	Completeness	Compactness	Löwenheim-Skolem
Standard	✓	✗	✗	✗ (upward)
Henkin	✗	✓	✓	(✓)

✓ = holds ✗ = does not hold

- Undecidability of validity & satisfiability in PA_2 and in the empty signature

Conclusion

Contributions: To the best of our knowledge, first mechanization of SOL.

- Formalized the following meta-theoretic properties:

Semantics	$\mathbb{N}$ Categorical	Completeness	Compactness	Löwenheim-Skolem
Standard	✓	✗	✗	✗ (upward)
Henkin	✗	✓	✓	(✓)

✓ = holds ✗ = does not hold

- Undecidability of validity & satisfiability in PA_2 and in the empty signature

Mechanization: $\sim$ 10,000 LOC overall

Conclusion

Contributions: To the best of our knowledge, first mechanization of SOL.

- Formalized the following meta-theoretic properties:

Semantics	$\mathbb{N}$ Categorical	Completeness	Compactness	Löwenheim-Skolem
Standard	✓	✗	✗	✗ (upward)
Henkin	✗	✓	✓	(✓)

✓ = holds ✗ = does not hold

- Undecidability of validity & satisfiability in PA_2 and in the empty signature

Mechanization: $\sim$ 10,000 LOC overall

- Formalization of categoricity worked relatively smoothly

Conclusion

Contributions: To the best of our knowledge, first mechanization of SOL.

- Formalized the following meta-theoretic properties:

Semantics	$\mathbb{N}$ Categorical	Completeness	Compactness	Löwenheim-Skolem
Standard	✓	✗	✗	✗ (upward)
Henkin	✗	✓	✓	(✓)

✓ = holds ✗ = does not hold

- Undecidability of validity & satisfiability in PA_2 and in the empty signature

Mechanization: $\sim$ 10,000 LOC overall

- Formalization of categoricity worked relatively smoothly
- Henkin reduction by far the most difficult part ($\sim$ 2,000 LOC)
  $\Rightarrow$ Especially handling of de Bruijn encoding challenging

Future Work

- Merge development into Coq Library of Undecidability Proofs [Forster et al., 2019b]

Future Work

- Merge development into Coq Library of Undecidability Proofs [Forster et al., 2019b]
- Connect with FOL completeness mechanization [Forster et al., 2021]

Future Work

- Merge development into Coq Library of Undecidability Proofs [Forster et al., 2019b]
- Connect with FOL completeness mechanization [Forster et al., 2021]
- Conservativity of PA_2 over PA_1

Future Work

- Merge development into Coq Library of Undecidability Proofs [Forster et al., 2019b]
- Connect with FOL completeness mechanization [Forster et al., 2021]
- Conservativity of PA_2 over PA_1
- Second-order set-theory & real analysis

Future Work

- Merge development into Coq Library of Undecidability Proofs [Forster et al., 2019b]
- Connect with FOL completeness mechanization [Forster et al., 2021]
- Conservativity of PA_2 over PA_1
- Second-order set-theory & real analysis
- Internal categoricity [Väänänen and Wang, 2012]

References i


Bauer, A. (2006).
First steps in synthetic computability theory.
Electronic Notes in Theoretical Computer Science, 155:5–31.
Proceedings of the 21st Annual Conference on Mathematical Foundations of Programming Semantics (MFPS XXI).


Forster, Y., Kirst, D., and Smolka, G. (2019a).
On synthetic undecidability in coq, with an application to the entscheidungsproblem.
In *Proceedings of the 8th ACM SIGPLAN International Conference on Certified Programs and Proofs*, CPP 2019, page 38–51, New York, NY, USA. Association for Computing Machinery.


Forster, Y., Kirst, D., and Wehr, D. (2021).
Completeness theorems for first-order logic analysed in constructive type theory: Extended version.
Journal of Logic and Computation, 31(1):112–151.


Forster, Y., Larchey-Wendling, D., Dudenhefner, A., Heiter, E., Kirst, D., Kunze, F., and Smolka, G. (2019b).
A coq library of undecidable problems.

References ii


Kirst, D. and Hermes, M. (2021).
Synthetic undecidability and incompleteness of first-order axiom systems in coq.
In *ITP*.


Kirst, D. and Larchey-Wendling, D. (2020).
Trakhtenbrot's theorem in coq: A constructive approach to finite model theory.
International Joint Conference on Automated Reasoning.


Kirst, D. and Smolka, G. (2017).
Categoricity results for second-order zf in dependent type theory.
In *ITP*.


Nour, K. and Raffalli, C. (2003).
Simple proof of the completeness theorem for second-order classical and intuitionistic logic by reduction to first-order mono-sorted logic.
Theoretical computer science, 308(1-3):227–237.


Shapiro, S. (1991).
***Foundations without foundationalism: A case for second-order logic*, volume 17.**
Clarendon Press.

References iii


Väänänen, J. and Wang, T. (2012).
Internal categoricity in arithmetic and set theory.
Notre Dame Journal of Formal Logic, 56.

Undecidability of Validity

$$\forall \mathcal{M}.\, \mathcal{M} \vDash \dot{\forall}\, f_0\, f_S\, f_+\, f_\times\, P_\equiv.\, \mathrm{PA}_2' \mathrel{\dot{\to}} \varphi'_{p,q}$$
$$\updownarrow$$
$$\forall \mathcal{M}.\, \mathcal{M} \vDash \mathrm{PA}_2 \to \mathcal{M} \vDash \varphi_{p,q}$$

Lemma

$p = q$ has a solution iff $\mathcal{M} \vDash \varphi_{p,q}$ for all models with $\mathcal{M} \vDash \mathrm{PA}_2$.

Proof.

$\to$: Two possible proofs:

- If $p = q$ has a solution, then $\mathbb{N} \vDash \varphi_{p,q}$. By categoricity it holds for all models of PA_2.
- Translate $p = q$ solution to $\mathcal{M}$ using a homomorphism $\mu : \mathbb{N} \to \mathcal{M}$.

$\leftarrow$: Instantiate $\mathcal{M}$ with standard model $\mathbb{N}$ to obtain $\mathbb{N} \vDash \varphi_{p,q}$.

Undecidability of Satisfiability

$$\exists \mathcal{M} \rho.\, \mathcal{M} \vDash_\rho \dot{\exists}\, f_0\, f_S\, f_+\, f_\times\, P_\equiv.\, \mathsf{PA}_2' \mathbin{\dot{\wedge}} \varphi'_{p,q}$$
$$\updownarrow$$
$$\exists \mathcal{M} \rho.\, \mathcal{M}, \rho \vDash \mathsf{PA}_2 \wedge \mathcal{M}, \rho \vDash \varphi_{p,q}$$

Lemma

$p = q$ has a solution iff there is a model $\mathcal{M} \vDash \mathsf{PA}_2$ and ρ such that $\mathcal{M} \vDash_\rho \varphi_{p,q}$.

Proof.

$\rightarrow$: If $p = q$ has a solution, then the standard model $\mathbb{N}$ fulfils $\mathbb{N} \vDash \varphi_{p,q}$.

$\leftarrow$: If $\mathcal{M}, \rho \vDash \varphi_{p,q}$ then also $\mathbb{N} \vDash \varphi_{p,q}$ by categoricity.

Note that categoricity was required here, whereas it is optional for validity.

Non-Enumerability

Theorem.

Enumerability of validity in PA_2 implies decidability of the halting problem under MP.

Proof.

- Enumerate H_{10} via $\mathrm{PA}_2 \vDash \varphi_{p,q}$.
- Enumerate $\overline{\mathrm{H}_{10}}$ via $\neg\mathrm{PA}_2 \vDash \varphi_{p,q} \;\leftrightarrow\; \mathrm{PA}_2 \vDash \dot{\neg}\varphi_{p,q}$ (Categoricity).

Yields decider via Post's theorem. □

Natural Deduction

$$\frac{A[\uparrow_p^n] \vdash_2 \varphi}{A \vdash_2 \dot{\forall}_p^n \varphi}\ \mathrm{AI_p} \qquad \frac{A \vdash_2 \dot{\forall}_p^n \varphi}{A \vdash_2 \varphi[P]}\ \mathrm{AE_p}$$

$$\frac{A \vdash_2 \varphi[P]}{A \vdash_2 \dot{\exists}_p^n \varphi}\ \mathrm{EI_p} \qquad \frac{A \vdash_2 \dot{\exists}_p^n \varphi \qquad A[\uparrow_p^n], \varphi \vdash_2 \psi[\uparrow_p^n]}{A \vdash_2 \psi}\ \mathrm{EE_p}$$

$$\frac{}{A \vdash_2 \dot{\exists}_p^n P.\, \dot{\forall} x_1 \ldots x_n.\, P(x_1, \ldots, x_2) \dot{\leftrightarrow} \varphi[\uparrow_p^n]}\ \mathrm{Compr}_p$$

Semantic Henkin Reduction

- Turn Henkin model $\mathcal{H}$ into first-order model $\mathcal{H}^\star$ with $D^\star := D \cup \mathbb{U}$ and $\mathsf{App}_n\,(x :: \boldsymbol{v}) := \mathsf{toPred}_n\,x\,(\mathsf{toIndi}\,\boldsymbol{v})$

$$\mathcal{H} \vDash_2 \varphi \;\leftrightarrow\; \mathcal{H}^\star \vDash_1 \varphi^\star$$

- Turn first-order model $\mathcal{M}$ into Henkin model $\mathcal{M}^\diamond$ with $D^\diamond := D$ and $\mathbb{U}$ induces by interpretation of App.

$$\mathcal{M} \vDash_1 \mathsf{Comprehension}^\star \;\to\; \mathcal{M}^\diamond \vDash_2 \varphi \;\leftrightarrow\; \mathcal{M} \vDash_1 \varphi^\star$$

Undecidability of Validity

$$\forall \mathcal{M}.\, \mathcal{M} \vDash \dot{\forall}\, f_0\, f_S\, f_+\, f_\times\, P_\equiv.\, \mathrm{PA}_2' \mathrel{\dot{\to}} \varphi'_{p,q}$$
$$\updownarrow$$
$$\forall \mathcal{M}.\, \mathcal{M} \vDash \mathrm{PA}_2 \to \mathcal{M} \vDash \varphi_{p,q}$$

Lemma

$p = q$ has a solution iff $\mathcal{M} \vDash \varphi_{p,q}$ for all models with $\mathcal{M} \vDash \mathrm{PA}_2$.

Proof.

$\to$: Two possible proofs:

- If $p = q$ has a solution, then $\mathbb{N} \vDash \varphi_{p,q}$. By categoricity it holds for all models of PA_2.
- Translate $p = q$ solution to $\mathcal{M}$ using a homomorphism $\mu : \mathbb{N} \to \mathcal{M}$.

$\leftarrow$: Instantiate $\mathcal{M}$ with standard model $\mathbb{N}$ to obtain $\mathbb{N} \vDash \varphi_{p,q}$.

Undecidability of Satisfiability

$$\exists \mathcal{M} \rho.\, \mathcal{M} \vDash_\rho \dot{\exists}\, f_0\, f_S\, f_+\, f_\times\, P_\equiv.\, \mathrm{PA}_2' \mathbin{\dot{\wedge}} \varphi'_{p,q}$$
$$\updownarrow$$
$$\exists \mathcal{M} \rho.\, \mathcal{M} \vDash \mathrm{PA}_2 \wedge \mathcal{M} \vDash_\rho \varphi_{p,q}$$

Lemma

$p = q$ has a solution iff there is a model $\mathcal{M} \vDash \mathrm{PA}_2$ and ρ such that $\mathcal{M} \vDash_\rho \varphi_{p,q}$.

Proof.

$\rightarrow$: If $p = q$ has a solution, then the standard model $\mathbb{N}$ fulfils $\mathbb{N} \vDash \varphi_{p,q}$.

$\leftarrow$: If $\mathcal{M} \vDash_\rho \varphi_{p,q}$ then also $\mathbb{N} \vDash \varphi_{p,q}$ by categoricity.

Note that categoricity was required here, whereas it is optional for validity.

Natural Deduction

$$\frac{A[\uparrow_f^n] \vdash \varphi}{A \vdash \dot{\forall}_f^n \varphi}\ \mathrm{AI_f} \qquad \frac{A \vdash \dot{\forall}_f^n \varphi}{A \vdash \varphi[f]}\ \mathrm{AE_f}$$

$$\frac{A \vdash \varphi[f]}{A \vdash \dot{\exists}_f^n \varphi}\ \mathrm{EI_f} \qquad \frac{A \vdash \dot{\exists}_f^n \varphi \qquad A[\uparrow_f^n], \varphi \vdash \psi[\uparrow_n]_f}{A \vdash \psi}\ \mathrm{EE_f}$$

$$\frac{}{\dot{\exists}_p^n P.\, \dot{\forall} x_1 \ldots x_n.\, P(x_1, \ldots, x_2) \dot{\leftrightarrow} \varphi[\uparrow_p^n]}\ \mathrm{Compr}$$

Backwards Translation

Define a backwards translation $\_^{\diamond} : \mathsf{form}_1(\Sigma_+) \to \mathsf{form}_2(\Sigma_{\mathsf{err}})$. For example

$$(\forall x.\, \mathsf{predApp}_0(x) \mathbin{\dot\wedge} \mathsf{predApp}_1(x, x))^{\diamond}$$

$$||$$

$$\forall x_i.\, \forall^0_p\, x^0_p.\, \forall^1_p\, x^1_p.\, x^0_p \mathbin{\dot\wedge} x^1_p(x_i)$$

$$(\mathsf{predApp}_1(f(x), y))^{\diamond} = \mathsf{Err}_1(y_i)$$

Special error symbol if first argument is not a variable

Completeness

Lemma

1. $A \vdash_1 \varphi \;\rightarrow\; A^{\diamond} \vdash_2^{\mathsf{err}} \varphi^{\diamond}$
2. $\vdash_2 \varphi^{\star\diamond} \dot{\leftrightarrow} \varphi$

$$\mathcal{T} \vDash_2 \varphi \;\longleftrightarrow\; \mathcal{T}^{\star}, \mathcal{C} \vDash_1 \varphi^{\star}$$

FOL Completeness [Forster et al., 2021]

LEM

$$\mathcal{T}^{\star}, \mathcal{C} \vdash_1 \varphi^{\star} \xrightarrow{(1)} \mathcal{T}^{\star\diamond}, \mathcal{C}^{\diamond} \vdash_2^{\mathsf{err}} \varphi^{\star\diamond}$$

Error symbol can occur in this derivation!

Remove error symbol using comprehension

$$\mathcal{T}^{\star\diamond}, \mathcal{C}^{\diamond} \vdash_2 \varphi^{\star\diamond} \xleftrightarrow{(2)} \mathcal{T}, \mathcal{C}^{\diamond} \vdash_2 \varphi \xrightarrow[\vdash_2 \text{ proves comprehension}]{} \mathcal{T} \vdash_2 \varphi$$

Internal Categoricity [Väänänen and Wang, 2012]

Consider a theory $\mathcal{T}$ depending on a single predicate symbol $\mathcal{P}$

$$\mathsf{Categ}(\mathcal{T}) := \dot{\forall} D_1 D_2 P_1 P_2 . \mathcal{T}(P_1)^{D_1} \dot{\to} \mathcal{T}(P_2)^{D_2} \dot{\to} \dot{\exists} \cong . \mathsf{Iso}(\cong, D_1, D_2, P_1, P_2)$$

where $\mathcal{T}(P_1)^{D_1}$ replaces $\mathcal{P}$ with the variable P_1 and guards all quantifiers with the domain predicate D_1.

- $\mathcal{T}$ is categorical iff $\vDash \mathsf{Categ}(\mathcal{T})$
- Provable in many cases (despite incompleteness), e.g. $\vdash \mathsf{Categ}(\mathsf{PA}_2)$.
  - $\Rightarrow$ Categoricity can be written and proven at the object level, without depending on any external set theory (or type theory in our case)