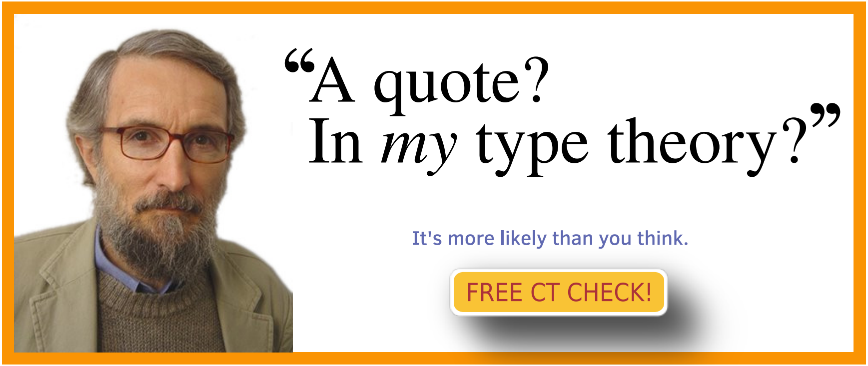

Pierre-Marie Pédrot

INRIA

2023/10/17

Today's Focus

Church's thesis!

All reasonable computational models are equivalent.

Today's Focus

[illegible]

All reasonable computational models are equivalent.

Today's Focus

~~[illegible]~~

All reasonable computational models are equivalent.

The internal Church thesis in a theory $\mathcal{T}$!

From within $\mathcal{T}$, "all functions $\mathbb{N} \to \mathbb{N}$ are computable".

Horrible Encodings Ahead

Let's fix a simple type theory $\mathcal{T}$ containing arithmetic.

$\rightsquigarrow$ One can define the (decidable) Turing predicate:

$$\frac{p : \mathbb{N} \qquad n : \mathbb{N} \qquad k : \mathbb{N}}{\mathsf{T}(p, n, k) : \mathsf{Prop}}$$

"$\mathsf{T}(p, n, k)$ holds iff the Turing machine p returns n in $\leq k$ steps."

Horrible Encodings Ahead

Let's fix a simple type theory $\mathcal{T}$ containing arithmetic.

$\rightsquigarrow$ One can define the (decidable) Turing predicate:

$$\frac{p : \mathbb{N} \qquad n : \mathbb{N} \qquad k : \mathbb{N}}{\mathsf{T}(p, n, k) : \mathsf{Prop}}$$

"$\mathsf{T}(p, n, k)$ holds iff the Turing machine p returns n in $\leq k$ steps."

[NB: for readability, I'll henceforth write $\mathbb{P} := \mathbb{N}$ to indicate numbers coding programs]

Horrible Encodings Ahead

Let's fix a simple type theory $\mathcal{T}$ containing arithmetic.

$\rightsquigarrow$ One can define the (decidable) Turing predicate:

$$\frac{p : \mathbb{N} \qquad n : \mathbb{N} \qquad k : \mathbb{N}}{\mathsf{T}(p, n, k) : \mathsf{Prop}}$$

"$\mathsf{T}(p, n, k)$ holds iff the Turing machine p returns n in $\leq k$ steps."

[NB: for readability, I'll henceforth write $\mathbb{P} := \mathbb{N}$ to indicate numbers coding programs]

$\rightsquigarrow$ We say that $f : \mathbb{N} \to \mathbb{N}$ is computed by $p : \mathbb{P}$, written $\mathbf{calc}\ f\ p$ when

$$\vdash_{\mathcal{T}} \forall n : \mathbb{N}.\, \exists k : \mathbb{N}.\, \mathsf{T}(p \bullet n, f\ n, k)$$

Horrible Encodings Ahead

Let's fix a simple type theory $\mathcal{T}$ containing arithmetic.

$\rightsquigarrow$ One can define the (decidable) Turing predicate:

$$\frac{p : \mathbb{N} \qquad n : \mathbb{N} \qquad k : \mathbb{N}}{\mathsf{T}(p, n, k) : \mathsf{Prop}}$$

"$\mathsf{T}(p, n, k)$ holds iff the Turing machine p returns n in $\leq k$ steps."

[NB: for readability, I'll henceforth write $\mathbb{P} := \mathbb{N}$ to indicate numbers coding programs]

$\rightsquigarrow$ We say that $f : \mathbb{N} \rightarrow \mathbb{N}$ is computed by $p : \mathbb{P}$, written $\mathbf{calc}\ f\ p$ when

$$\vdash_{\mathcal{T}} \forall n : \mathbb{N}.\, \exists k : \mathbb{N}.\, \mathsf{T}(p \bullet n, f\ n, k)$$

Internal CT

$\mathcal{T}$ validates CT if $\vdash_{\mathcal{T}} \forall f : \mathbb{N} \rightarrow \mathbb{N}.\, \exists p : \mathbb{P}.\, \mathbf{calc}\ f\ p$

Alonzo in Maŝinmondo

CT is a weird principle!

- Implies a mechanical world
- A staple of Russian constructivism
- In presence of choice, incompatible with funext
- In presence of choice, incompatible with classical logic

CT is a weird principle!

- Implies a mechanical world
- A staple of Russian constructivism
- In presence of choice, incompatible with funext
- In presence of choice, incompatible with classical logic

A fleeting panic

Is it actually consistent?

Alonzo in Maŝinmondo

CT is a weird principle!

- Implies a mechanical world
- A staple of Russian constructivism
- In presence of choice, incompatible with funext
- In presence of choice, incompatible with classical logic

A fleeting panic

Is it actually consistent?

Ja.

(Mumble something about The Effective Topos™ being a model of HOL + CT.)

We Need to Go Deeper

A Legitimate Question

"Can we extend Martin-Löf's Type Theory with CT?"

We Need to Go Deeper

A Legitimate Question

"Can we extend Martin-Löf's Type Theory with CT?"

An Even More Legitimate Question

"Why would I do that?"

We Need to Go Deeper

A Legitimate Question

"Can we extend Martin-Löf's Type Theory with CT?"

An Even More Legitimate Question

"Why would I do that?"

- This watch does not smell of mustard.
- Simple type theory is cool, but a bit old-fashioned and limited
- In MLTT, functions are *already* programs
- MLTT + CT is the foundation for **synthetic computability**

I think, Therefore I merely am

In dependent type theories, existing is a complex matter

$\Sigma x : A.\, B$	v.s.	$\exists x : A.\, B$
actual existence		mere existence
proof relevant		proof-irrelevant
choice built-in		no choice *a priori*

I think, Therefore I merely am

In dependent type theories, existing is a complex matter

$\Sigma x : A.\, B$	v.s.	$\exists x : A.\, B$
actual existence		mere existence
proof relevant		proof-irrelevant
choice built-in		no choice *a priori*

We have not one, but *two* theses.

$$\begin{array}{lcl} \mathsf{CT}_{\exists} & := & \Pi(f : \mathbb{N} \rightarrow \mathbb{N}).\, \exists p : \mathbb{P}.\, \mathbf{calc}\ f\ p \\ \mathsf{CT}_{\Sigma} & := & \Pi(f : \mathbb{N} \rightarrow \mathbb{N}).\, \Sigma p : \mathbb{P}.\, \mathbf{calc}\ f\ p \end{array}$$

Due to the lack of choice, $\mathsf{CT}_{\exists}$ is known to be consistent in MLTT.

(The Effective Topos™)

I think, Therefore I merely am

In dependent type theories, existing is a complex matter

$\Sigma x : A.\, B$	v.s.	$\exists x : A.\, B$
actual existence		mere existence
proof relevant		proof-irrelevant
choice built-in		no choice *a priori*

We have not one, but *two* theses.

$$\begin{array}{lcl} \mathsf{CT}_{\exists} & := & \Pi(f : \mathbb{N} \to \mathbb{N}).\, \exists p : \mathbb{P}.\, \mathbf{calc}\ f\ p \\ \mathsf{CT}_{\Sigma} & := & \Pi(f : \mathbb{N} \to \mathbb{N}).\, \Sigma p : \mathbb{P}.\, \mathbf{calc}\ f\ p \end{array}$$

Due to the lack of choice, $\mathsf{CT}_{\exists}$ is known to be consistent in MLTT.

(The Effective Topos™)

Dually, CT_{Σ} is the hallmark of weird crap going on

Second-hand "Quotes" from Anonymous Experts**

M.E. (Birmingham)

"MLTT is obviously inconsistent with CT_Σ"

"I believe that MLTT cannot validate CT_Σ"

T.S. (Darmstadt)

** All these quotes are a pure work of fiction. Serving suggestion. May contain phthalates.

Somebody is Wrong on the Internet

Are you seriously kidding me?

Somebody is Wrong on the Internet

Are you seriously kidding me?

- In MLTT, functions are **already** frigging programs!
- CT_Σ holds externally, it's called extraction (duh)

$$\text{for all} \quad \vdash f : \mathbb{N} \to \mathbb{N} \quad \text{there is} \quad \vdash p : \mathbb{P} \quad \text{s.t.} \quad \vdash \mathbf{calc}\ f\ p$$

- It is hence **obvious** that CT_Σ is compatible with MLTT
- We just have to handle those pesky **open** terms!

Somebody is Wrong on the Internet

Are you seriously kidding me?

- In MLTT, functions are **already** frigging programs!
- CT_Σ holds externally, it's called extraction (duh)

$$\text{for all} \quad \vdash f : \mathbb{N} \to \mathbb{N} \quad \text{there is} \quad \vdash p : \mathbb{P} \quad \text{s.t.} \quad \vdash \mathbf{calc}\ f\ p$$

- It is hence **obvious** that CT_Σ is compatible with MLTT
- We just have to handle those pesky **open** terms!

Only one way out: prove that I am right!

- Define an extension of MLTT proving CT_Σ
- Prove it's consistent / canonical / strongly normalizing / ...
- Formalize this in Coq otherwise nobody believes you

Somebody is Wrong on the Internet

Are you seriously kidding me?

- In MLTT, functions are **already** frigging programs!
- CT_Σ holds externally, it's called extraction (duh)

$$\text{for all} \quad \vdash f : \mathbb{N} \to \mathbb{N} \quad \text{there is} \quad \vdash p : \mathbb{P} \quad \text{s.t.} \quad \vdash \mathbf{calc}\ f\ p$$

- It is hence **obvious** that CT_Σ is compatible with MLTT
- We just have to handle those pesky **open** terms!

Only one way out: prove that I am right!

- Define an extension of MLTT proving CT_Σ
- Prove it's consistent / canonical / strongly normalizing / ...
- Formalize this in Coq otherwise nobody believes you

Spoiler alter: we will sketch that in the rest of the talk.

"MLTT"

We define "MLTT" as the extension of MLTT with two new primitives.

$$M, N := \ldots \mid \text{♀}\ M \mid \text{♀}\ M\ N$$

$$\frac{\Gamma \vdash M : \mathbb{N} \rightarrow \mathbb{N}}{\Gamma \vdash \text{♀}\ M : \mathbb{P}} \qquad \frac{\Gamma \vdash M : \mathbb{N} \rightarrow \mathbb{N} \qquad \Gamma \vdash N : \mathbb{N}}{\Gamma \vdash \text{♀}\ M\ N : \texttt{eval}\ (\text{♀}\ M)\ N\ (M\ N)}$$

where

$$\begin{array}{lcl} \texttt{eval} & : & \mathbb{P} \rightarrow \mathbb{N} \rightarrow \mathbb{N} \rightarrow \square \\ \texttt{eval}\ P\ N\ V & \sim & \text{program } P \text{ applied to } N \text{ normalizes to } V \end{array}$$

“MLTT”

We define “MLTT” as the extension of MLTT with two new primitives.

$$M, N := \ldots \mid \text{⚲}\ M \mid \text{♀}\ M\ N$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash \text{⚲}\ M : \mathbb{P}} \qquad \frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N : \mathbb{N}}{\Gamma \vdash \text{♀}\ M\ N : \texttt{eval}\ (\text{⚲}\ M)\ N\ (M\ N)}$$

where

$\texttt{eval}$	:	$\mathbb{P} \to \mathbb{N} \to \mathbb{N} \to \square$
$\texttt{eval}\ P\ N\ V$	$\sim$	program P applied to N normalizes to V

The system is parameterized by a *computation model*, given by:

- A meta-function $\ulcorner\cdot\urcorner : \texttt{term} \Rightarrow \mathbf{N}$ (your favourite Gödel numbering)

“MLTT”

We define “MLTT” as the extension of MLTT with two new primitives.

$$M, N := \ldots \mid \text{♀}\ M \mid \text{♀}\ M\ N$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash \text{♀}\ M : \mathbb{P}} \qquad \frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N : \mathbb{N}}{\Gamma \vdash \text{♀}\ M\ N : \texttt{eval}\ (\text{♀}\ M)\ N\ (M\ N)}$$

where

$$\begin{array}{lcl} \texttt{eval} & : & \mathbb{P} \to \mathbb{N} \to \mathbb{N} \to \square \\ \texttt{eval}\ P\ N\ V & \sim & \text{program } P \text{ applied to } N \text{ normalizes to } V \end{array}$$

The system is parameterized by a *computation model*, given by:

- A meta-function $\ulcorner \cdot \urcorner : \texttt{term} \Rightarrow \mathbf{N}$ (your favourite Gödel numbering)
- An MLTT function $\vdash \texttt{run} : \mathbb{P} \to \mathbb{N} \to \mathfrak{P}(\mathbb{N})$

where $\mathfrak{P}(A) := \mathbb{N} \to \texttt{option}\ A$ is the partiality monad

and eval is derived from run through standard combinators

The Hard Part

What is the hard part?

The Hard Part

What is the hard part?

Conversion!

What is the hard part?

Conversion!

$$\frac{\Gamma \vdash M : B \qquad \Gamma \vdash A \equiv B}{\Gamma \vdash M : A}$$

In MLTT the type system embeds the runtime.

The Hard Part

What is the hard part?

Conversion!

$$\frac{\Gamma \vdash M : B \qquad \Gamma \vdash A \equiv B}{\Gamma \vdash M : A}$$

In MLTT the type system embeds the runtime.

We need to ensure that convertible terms are quoted to the same number.

Remember that CT_Σ is inconsistent with funext.

Thankfully conversion is intensional in MLTT...

Naive Solution

We need to ensure that convertible terms are quoted to the same number.

Naive Solution

We need to ensure that convertible terms are quoted to the same number.

Assume we can magically choose one representative per convertibility class.

$$\Gamma \vdash M \equiv N : \mathbb{N} \to \mathbb{N} \quad \text{iff} \quad \ulcorner \varepsilon(M) \urcorner = \ulcorner \varepsilon(N) \urcorner$$

Unfortunately, this is not going to be stable by substitution.

$$\varepsilon(M\{x := N\}) \neq \varepsilon(M)\{x := \varepsilon(N)\}$$

Immediate breakage of conversion!

The major insight for “MLTT”

Open terms are a lie! It's a conspiracy from Big Variable!

The major insight for “MLTT”

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

The major insight for “MLTT”

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

ϙ and Ϙ will only compute on (deep normal) **closed** terms

The major insight for “MLTT”

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

♀ and ♀ will only compute on (deep normal) **closed** terms

$$\frac{\Gamma \vdash M \equiv N : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash \text{♀}\ M \equiv \text{♀}\ N : \mathbb{P}} \qquad \frac{\Gamma \vdash M \equiv M' : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N \equiv N' : \mathbb{N}}{\Gamma \vdash \text{♀}\ M\ N \equiv \text{♀}\ M'\ N' : \texttt{eval}\ (\text{♀}\ M)\ N\ (M\ N)}$$

The major insight for "MLTT"

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

⚲ and ⚲ will only compute on (deep normal) **closed** terms

$$\frac{\Gamma \vdash M \equiv N : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash ⚲\ M \equiv ⚲\ N : \mathbb{P}} \qquad \frac{\Gamma \vdash M \equiv M' : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N \equiv N' : \mathbb{N}}{\Gamma \vdash ⚲\ M\ N \equiv ⚲\ M'\ N' : \texttt{eval}\ (⚲\ M)\ N\ (M\ N)}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash ⚲\ M \equiv \ulcorner M \urcorner : \mathbb{P}}$$

The major insight for “MLTT”

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

♀ and ♀ will only compute on (deep normal) closed terms

$$\frac{\Gamma \vdash M \equiv N : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash \text{♀}\ M \equiv \text{♀}\ N : \mathbb{P}} \qquad \frac{\Gamma \vdash M \equiv M' : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N \equiv N' : \mathbb{N}}{\Gamma \vdash \text{♀}\ M\ N \equiv \text{♀}\ M'\ N' : \texttt{eval}\ (\text{♀}\ M)\ N\ (M\ N)}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash \text{♀}\ M \equiv \lceil M \rceil : \mathbb{P}}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf} \qquad n \in \mathbf{N} \qquad \Gamma \vdash P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}{\Gamma \vdash \text{♀}\ M\ \overline{n} \equiv P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}$$

The major insight for "MLTT"

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

⚲ and ♀ will only compute on (deep normal) **closed** terms

$$\frac{\Gamma \vdash M \equiv N : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash ⚲\ M \equiv ⚲\ N : \mathbb{P}} \qquad \frac{\Gamma \vdash M \equiv M' : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N \equiv N' : \mathbb{N}}{\Gamma \vdash ♀\ M\ N \equiv ♀\ M'\ N' : \texttt{eval}\ (⚲\ M)\ N\ (M\ N)}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash ⚲\ M \equiv \lceil M \rceil : \mathbb{P}}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf} \qquad n \in \mathbf{N} \qquad \Gamma \vdash P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}{\Gamma \vdash ♀\ M\ \overline{n} \equiv P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}$$

This One Weird Trick

Closed terms are stable by substitution!

The major insight for “MLTT”

OPEN TERMS ARE A LIE! IT'S A CONSPIRACY FROM BIG VARIABLE!

(Source: $\mathbb{X}$.)

⚲ and ♀ will only compute on (deep normal) **closed** terms

$$\frac{\Gamma \vdash M \equiv N : \mathbb{N} \to \mathbb{N}}{\Gamma \vdash ⚲\ M \equiv ⚲\ N : \mathbb{P}} \qquad \frac{\Gamma \vdash M \equiv M' : \mathbb{N} \to \mathbb{N} \qquad \Gamma \vdash N \equiv N' : \mathbb{N}}{\Gamma \vdash ♀\ M\ N \equiv ♀\ M'\ N' : \texttt{eval}\ (⚲\ M)\ N\ (M\ N)}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash ⚲\ M \equiv \ulcorner M \urcorner : \mathbb{P}}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf} \qquad n \in \mathbf{N} \qquad \Gamma \vdash P : \texttt{eval}\ \ulcorner M \urcorner\ \overline{n}\ (M\ \overline{n})}{\Gamma \vdash ♀\ M\ \overline{n} \equiv P : \texttt{eval}\ \ulcorner M \urcorner\ \overline{n}\ (M\ \overline{n})}$$

This One Weird Trick

Closed terms are stable by substitution!

(Some additional technicalities to validate η-laws.)

The Secret Sauce

These functions return hereditarily positive types (aka Σ^0_1 formulae)

The Secret Sauce

These functions return hereditarily positive types (aka Σ^0_1 formulae)

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash \text{♀}\ M \equiv \lceil M \rceil : \mathbb{P}}$$

The Secret Sauce

These functions return hereditarily positive types (aka Σ^0_1 formulae)

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash \text{♀}\ M \equiv \lceil M \rceil : \mathbb{P}}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf} \qquad n \in \mathbf{N} \qquad \Gamma \vdash P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}{\Gamma \vdash \text{♀}\ M\ \overline{n} \equiv P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}$$

$$\begin{array}{lcl} \texttt{eval} & : & \mathbb{P} \to \mathbb{N} \to \mathbb{N} \to \square \\ \texttt{eval}\ f\ n\ v & := & \Sigma k : \mathbb{N}.\, \texttt{step}\ k\ (\texttt{run}\ f\ n)\ v \end{array}$$

$$\begin{array}{lcl} \texttt{step} & : & \mathbb{N} \to (\mathbb{N} \to \texttt{option}\ \mathbb{N}) \to \mathbb{N} \to \square \\ \texttt{step}\ \mathsf{O}\ p\ v & := & p\ \mathsf{O} = \texttt{Some}\ v \\ \texttt{step}\ (\mathsf{S}\ k)\ p\ v & := & (p\ (\mathsf{S}\ k) = \texttt{None}) \times (\texttt{eval}\ k\ (p \circ \mathsf{S})\ v) \end{array}$$

The Secret Sauce

These functions return hereditarily positive types (aka Σ^0_1 formulae)

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf}}{\Gamma \vdash \text{♀}\ M \equiv \lceil M \rceil : \mathbb{P}}$$

$$\frac{\Gamma \vdash M : \mathbb{N} \to \mathbb{N} \qquad M \text{ closed dnf} \qquad n \in \mathbf{N} \qquad \Gamma \vdash P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}{\Gamma \vdash \text{♀}\ M\ \overline{n} \equiv P : \texttt{eval}\ \lceil M \rceil\ \overline{n}\ (M\ \overline{n})}$$

$$\begin{array}{lcl} \texttt{eval} & : & \mathbb{P} \to \mathbb{N} \to \mathbb{N} \to \square \\ \texttt{eval}\ f\ n\ v & := & \Sigma k : \mathbb{N}.\ \texttt{step}\ k\ (\texttt{run}\ f\ n)\ v \end{array}$$

$$\begin{array}{lcl} \texttt{step} & : & \mathbb{N} \to (\mathbb{N} \to \texttt{option}\ \mathbb{N}) \to \mathbb{N} \to \square \\ \texttt{step}\ \mathsf{O}\ p\ v & := & p\ \mathsf{O} = \texttt{Some}\ v \\ \texttt{step}\ (\mathsf{S}\ k)\ p\ v & := & (p\ (\mathsf{S}\ k) = \texttt{None}) \times (\texttt{eval}\ k\ (p \circ \mathsf{S})\ v) \end{array}$$

These types have canonical “absolute” values!

The Basic Model

A variant of Abel's style NbE model in (small) IR

The Basic Model

A variant of Abel's style NbE model in (small) IR

- Type formation is defined inductively: $\Gamma \Vdash A$

$$\frac{A \Rightarrow^*_{\mathsf{wh}} \mathbb{N}}{\mathfrak{r}_{\mathbb{N}} : \Gamma \Vdash A} \qquad \frac{A \Rightarrow^*_{\mathsf{wh}} \Pi(x : X).\, Y \qquad p : \Gamma \Vdash X \qquad q : \Gamma, x : X \Vdash Y}{\mathfrak{r}_{\Pi}\ p\ q : \Gamma \Vdash A} \qquad \ldots$$

The Basic Model

A variant of Abel's style NbE model in (small) IR

- Type formation is defined inductively: $\Gamma \Vdash A$

$$\frac{A \Rightarrow^*_{\mathsf{wh}} \mathbb{N}}{\mathfrak{r}_{\mathbb{N}} : \Gamma \Vdash A} \qquad \frac{A \Rightarrow^*_{\mathsf{wh}} \Pi(x : X).\, Y \qquad p : \Gamma \Vdash X \qquad q : \Gamma, x : X \Vdash Y}{\mathfrak{r}_{\Pi}\ p\ q : \Gamma \Vdash A} \qquad \ldots$$

- Term typedness $\Gamma \Vdash M : A \mid p$ is defined by recursion on $p : \Gamma \Vdash A$

$$\begin{array}{lcl} \Gamma \Vdash M : \mathbb{N} \mid \mathfrak{r}_{\mathbb{N}} & := & \Gamma \Vdash M \in \mathbb{N} \\ \Gamma \Vdash M : \Pi(x : A).\, B \mid \mathfrak{r}_{\Pi}\ p\ q & := & \\ \quad \Pi(\rho : \Delta \leq \Gamma).\, (\Delta \Vdash a : A\langle\rho\rangle \mid p) \to \Delta \Vdash M\langle\rho\rangle\ a : B\{\rho, a\} \mid q & & \end{array}$$

$$\frac{M \Rightarrow^*_{\mathsf{wh}} \mathsf{O}}{\Gamma \Vdash M \in \mathbb{N}} \qquad \frac{M \Rightarrow^*_{\mathsf{wh}} \mathsf{S}\ N \qquad \Gamma \Vdash N \in \mathbb{N}}{\Gamma \Vdash M \in \mathbb{N}} \qquad \frac{\Gamma \vdash n : \mathbb{N} \qquad \mathsf{wne}\,(n)}{\Gamma \Vdash n \in \mathbb{N}}$$

A variant of Abel's style NbE model in (small) IR

- Type formation is defined inductively: $\Gamma \Vdash A$

$$\frac{A \Rightarrow^*_{\mathsf{wh}} \mathbb{N}}{\mathfrak{r}_{\mathbb{N}} : \Gamma \Vdash A} \qquad \frac{A \Rightarrow^*_{\mathsf{wh}} \Pi(x : X).\, Y \qquad p : \Gamma \Vdash X \qquad q : \Gamma, x : X \Vdash Y}{\mathfrak{r}_{\Pi}\ p\ q : \Gamma \Vdash A} \qquad \ldots$$

- Term typedness $\Gamma \Vdash M : A \mid p$ is defined by recursion on $p : \Gamma \Vdash A$

$$\begin{array}{lcl} \Gamma \Vdash M : \mathbb{N} \mid \mathfrak{r}_{\mathbb{N}} & := & \Gamma \Vdash M \in \mathbb{N} \\ \Gamma \Vdash M : \Pi(x : A).\, B \mid \mathfrak{r}_{\Pi}\ p\ q & := & \\ \quad \Pi(\rho : \Delta \leq \Gamma).\, (\Delta \Vdash a : A\langle\rho\rangle \mid p) \to \Delta \Vdash M\langle\rho\rangle\ a : B\{\rho, a\} \mid q & & \end{array}$$

$$\frac{M \Rightarrow^*_{\mathsf{wh}} \mathsf{O}}{\Gamma \Vdash M \in \mathbb{N}} \qquad \frac{M \Rightarrow^*_{\mathsf{wh}} \mathsf{S}\ N \qquad \Gamma \Vdash N \in \mathbb{N}}{\Gamma \Vdash M \in \mathbb{N}} \qquad \frac{\Gamma \vdash n : \mathbb{N} \qquad \mathsf{wne}\,(n)}{\Gamma \Vdash n \in \mathbb{N}}$$

(ditto for conversion) (+ second layer of *validity* aka closure by substitution)

Some Dust under the Rug

"MLTT" is reduction-free. I didn't define properly reduction!

- For the MLTT fragment, weak-head reduction is standard.
- Deep reduction is just iterated weak-head reduction.
- In particular, it is deterministic (critical!)

Some Dust under the Rug

"MLTT" is reduction-free. I didn't define properly reduction!

- For the MLTT fragment, weak-head reduction is standard.
- Deep reduction is just iterated weak-head reduction.
- In particular, it is deterministic (critical!)
- Reduction for ⚲ is obvious

Some Dust under the Rug

"MLTT" is reduction-free. I didn't define properly reduction!

- For the MLTT fragment, weak-head reduction is standard.
- Deep reduction is just iterated weak-head reduction.
- In particular, it is deterministic (critical!)
- Reduction for **Ϙ** is obvious
- Only tricky case is the rule for Ϙ: basically compute the unique fuel

$$\frac{M \text{ closed, dnf} \qquad k \text{ smallest integer s.t.} \qquad \mathtt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k} \Downarrow \mathsf{Some}\ \overline{v}}{\text{Ϙ}\ M\ \overline{n} \Rightarrow_{\mathsf{wh}} (\overline{k}, \mathtt{refl}, \ldots, \mathtt{refl})}$$

Reminder: $\mathtt{eval}\ p\ n\ v := \Sigma k : \mathbb{N}. (\mathtt{run}\ p\ n\ \mathsf{O} = \mathsf{None}) \times \ldots \times (\mathtt{run}\ p\ n\ \overline{k-1} = \mathsf{None}) \times (\mathtt{run}\ p\ n\ \overline{k} = \mathsf{Some}\ \overline{v})$

Comparison with standard NbE

Type interpretation unchanged

- No funny business with effects or whatnot
- In particular we have the same canonicity properties

Comparison with standard NbE

Type interpretation unchanged

- No funny business with effects or whatnot
- In particular we have the same canonicity properties

Differences with Abel's model

$\rightsquigarrow$ annotate reducibility proofs with deep normalization

$$\Gamma \Vdash M : A \mid p_A \qquad \text{implies} \qquad M \Downarrow_{\text{deep}} M_0 \quad \text{with} \quad \Gamma \vdash M \equiv M_0 : A$$

$\rightsquigarrow$ normal / neutral terms generalized into deep and weak-head variants

$\rightsquigarrow$ extend neutrals to contain quotes blocked on open terms

$$\frac{\mathsf{dnf}\,(M) \qquad M \text{ not closed}}{\mathsf{wne}\,(\text{♀}\ M)} \qquad \frac{\mathsf{dnf}\,(M) \qquad \mathsf{dnf}\,(N) \qquad M \text{ or } N \text{ not closed}}{\mathsf{wne}\,(\text{♀}\ M\ N)}$$

Comparison with standard NbE

Type interpretation unchanged

- No funny business with effects or whatnot
- In particular we have the same canonicity properties

Differences with Abel's model

$\rightsquigarrow$ annotate reducibility proofs with deep normalization

$$\Gamma \Vdash M : A \mid p_A \quad \text{implies} \quad M \Downarrow_{\text{deep}} M_0 \quad \text{with} \quad \Gamma \vdash M \equiv M_0 : A$$

$\rightsquigarrow$ normal / neutral terms generalized into deep and weak-head variants

$\rightsquigarrow$ extend neutrals to contain quotes blocked on open terms

$$\frac{\text{dnf}\,(M) \qquad M \text{ not closed}}{\text{wne}\,(♀\ M)} \qquad \frac{\text{dnf}\,(M) \qquad \text{dnf}\,(N) \qquad M \text{ or } N \text{ not closed}}{\text{wne}\,(♀\ M\ N)}$$

... and that's about it.

The Theorems

We say that the computation model $(\lceil\cdot\rceil, \mathtt{run})$ is adequate when:
for all $M \in \mathtt{term}$ and $n, r \in \mathbf{N}$, $M\,\overline{n} \Downarrow_{\mathsf{deep}}\ \overline{r}$ implies there is $k \in \mathbf{N}$ s.t.

- $\mathtt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k} \Downarrow_{\mathsf{deep}}$ Some $\overline{r}$
- $\mathtt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k'} \Downarrow_{\mathsf{deep}}$ None   for all $k' < k$

The Theorems

We say that the computation model $(\lceil \cdot \rceil, \texttt{run})$ is adequate when:
for all $M \in \texttt{term}$ and $n, r \in \mathbf{N}$, $M\,\overline{n} \Downarrow_{\mathsf{deep}} \overline{r}$ implies there is $k \in \mathbf{N}$ s.t.

- $\texttt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k} \Downarrow_{\mathsf{deep}}$ Some $\overline{r}$
- $\texttt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k'} \Downarrow_{\mathsf{deep}}$ None for all $k' < k$

Theorem

If the model is adequate, the logical relation is sound and complete.

In particular, "MLTT" is consistent, enjoys canonicity and normalization.

The Theorems

We say that the computation model $(\lceil\cdot\rceil, \texttt{run})$ is adequate when:
for all $M \in \texttt{term}$ and $n, r \in \mathbf{N}$, $M\,\overline{n} \Downarrow_{\mathsf{deep}} \overline{r}$ implies there is $k \in \mathbf{N}$ s.t.

- $\texttt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k} \Downarrow_{\mathsf{deep}}$ Some $\overline{r}$
- $\texttt{run}\ \lceil M \rceil\ \overline{n}\ \overline{k'} \Downarrow_{\mathsf{deep}}$ None   for all $k' < k$

Theorem

If the model is adequate, the logical relation is sound and complete.

In particular, "MLTT" is consistent, enjoys canonicity and normalization.

Theorem (It's written on the can)

"MLTT" validates CT_Σ.

Formalization

Based on Adjedj et al. Coq implementation (using small IR)

MLTT $+$ ⚲ fully formalized in Coq

The exact theory contains one universe, Π / Σ types with η-laws, $\bot$, $\mathbb{N}$, $\mathbf{Id}$

The infrastructure / non-trivial parts are behind me (deep reduction!)

Formalization

Based on Adjedj et al. Coq implementation (using small IR)

MLTT $+$ ⚲ fully formalized in Coq

The exact theory contains one universe, Π / Σ types with η-laws, $\bot$, $\mathbb{N}$, $\mathbf{Id}$

The infrastructure / non-trivial parts are behind me (deep reduction!)

Adding ♀ relies on the same ingredients

No expected surprise, just tedious proofs on untyped syntax

Nightmare stuff I'm not gonna prove: the existence of adequate models

Conclusion

- $\mathrm{MLTT} + \mathrm{CT}_\Sigma$ is obviously consistent
- The model is a trivial adaptation of standard NbE models
- Open terms do not exist. I have met them.
- Partially proved in Coq
- I must be missing something from our anonymous experts

Thanks for your attention.