On the Computation of Proof Terms in Homotopical Type Theory

M1 Internship Defense

Johann Rosain
Supervised by: Thierry Coquand

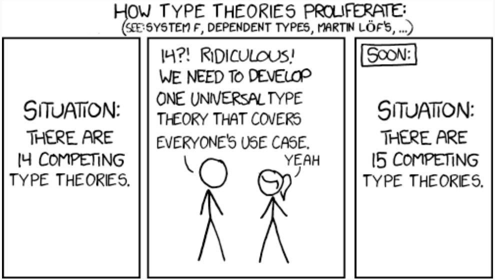
Lyon, Sept. 4, 2024

Computer Science Department
ENS Lyon
France

Yet Another Type Theory...

Did you say hot? No, I said HoTT!

- Foundation of mathematics
- Basis of formal systems
- Implementation in Coq, Agda, ...



Original image: `https://xkcd.com/927/`

HoTT is not Weird, it's Special

HoTT's *raison d'être*

Can be used to formalize weird things

Why Would *you* Want This? (maybe)

- Isomorphic structures are equal!! ☺
- Formalization of "properties up to isomorphism"
- Everything is (secretly) geometry

Why Would *I* Want This?

Having Fun While Working

- We can compute fun things!
- For instance: the number of groups of finite order

There's Always a "but"

- Very inefficient computations
- Very slow (hours?) to yield "1" with groups of order… 2 (duh)

Motivations

Our Goal

Find the reason(s) that make(s) it so bad.

Our Tool

postt, experimental type-checker s.t. HoTT computes (+ analysis).

Our Contributions

- A start of standard library for postt (impl.)
- Structures finiteness up to isomorphism (impl.)
- Analysis of the proof with (semi)groups

Dependent Types

$$\lambda x.t : \prod_{x:\,A} B(x) \qquad (x,p) : \sum_{x:\,A} P(x)$$

think of as $\forall x, B(x)$ — think of as $\exists x, P(x)$ + explicit witness

$A \to B$ shortcut for $\prod_{(\_:\,A)} B$ and $A \times B$ for $\sum_{(\_:\,A)} B$.

Inductive Types

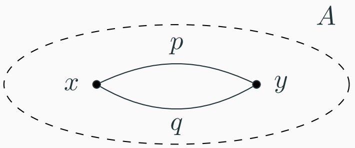
$\mathsf{inl}(x), \mathsf{inr}(y) : A + B$ — $\star : \mathbf{1}$, $\mathbf{0}$

think of as $A \vee B$ — think of as $\top$, $\bot$

$0 : \mathbb{N}, \mathsf{suc}_{\mathbb{N}}(n) : \mathbb{N}$ unary encoding of integers

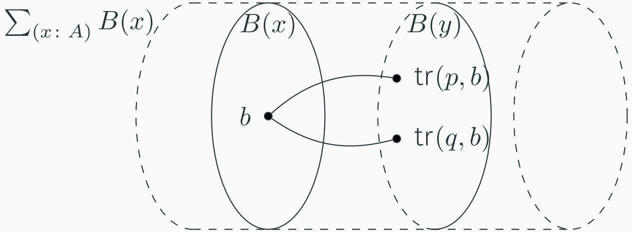
Identity Types

- Defined inductively by: $\mathsf{refl}_x : x =_A x$
- *Multiple* proofs of identity



$p, q : x =_A y$

Often-used operation: transport



The Finiteness we Choose

Standard Finite Types

$$\mathsf{Fin}_0 :\equiv \mathbf{0}$$

$$\mathsf{Fin}_{\mathsf{suc}_{\mathbb{N}}(n)} :\equiv \mathsf{Fin}_n + \mathbf{1}$$

i.e., there are n elements in Fin_n.

Equivalence

$A \simeq B$ if back-and-forth maps $f : A \to B, g, h : B \to A$ s.t.

$$f(g(x)) = x \qquad h(f(x)) = x$$

The Finiteness without Choice

Propositional Truncation

$\| A \|$: prop. trunc. of A, $\omega : \| A \|$ is an *undefined inhabitant* of A.

- $\left\| \sum_{(x : A)} P(x) \right\|$ is $\exists x, P(x)$ *without* explicit witness
- We thus write $\exists x, P(x)$ for $\left\| \sum_{(x : A)} P(x) \right\|$
- We only care about the fact that the type is inhabited

Finite Type

$\text{is-finite}(A) :\equiv \sum_{(k : \mathbb{N})} \| \text{Fin}_k \simeq A \|$

Another Example: Surjectivity

$\text{is-surj}(f) :\equiv \prod_{(y : B)} \exists x, f(x) = y$

(Finally) Our First Proof

Decidable Type

A is decidable if $d\colon A + \neg A$.

(Finally) Our First Proof

Decidable Type

A is decidable if $d\colon A + \neg A$.

Key Theorem 1: Finite Codomain

Let $f : A \to B$ a surjective function and A finite. Then B is finite whenever its equality is decidable.

Proof.

- Prop. is shown: get surjective map $g : \mathsf{Fin}_k \to B$.
- By induction on k . If $k \equiv 0$, then B is empty.
- $k > 0$: decide whether $g(\mathsf{inr}(\star))$ has more than 1 preimage
- If not, the induction hypothesis is enough.
- Otherwise, take n yielded by the induction hypothesis on B without $g(\mathsf{inr}(\star))$ and return $n + 1$

(Finally) Our First Proof

Decidable Type

A is decidable if $d\colon A + \neg A$.

Key Theorem 1: Finite Codomain

Let $f : A \to B$ a surjective function and A finite. Then B is finite whenever its equality is decidable.

Proof.

- Prop. is shown: get surjective map $g : \mathsf{Fin}_k \to B$.
- By induction on k (k steps). If $k \equiv 0$, then B is empty.
- $k > 0$: decide whether $g(\mathsf{inr}(\star))$ has more than 1 preimage
- If not, the induction hypothesis is enough.
- Otherwise, take n yielded by the induction hypothesis on B without $g(\mathsf{inr}(\star))$ and return $n + 1$

Let's take a look at the complexity

(Finally) Our First Proof

Decidable Type

A is decidable if $d \colon A + \neg A$.

Key Theorem 1: Finite Codomain

Let $f : A \to B$ a surjective function and A finite. Then B is finite whenever its equality is decidable.

Proof.

- Prop. is shown: get surjective map $g : \mathsf{Fin}_k \to B$.
- By induction on k (k steps). If $k \equiv 0$, then B is empty.
- $k > 0$: decide whether $g(\mathsf{inr}(\star))$ has more than 1 preimage (k values checked, d complexity of one check)
- If not, the induction hypothesis is enough.
- Otherwise, take n yielded by the induction hypothesis on B without $g(\mathsf{inr}(\star))$ and return $n + 1$

Let's take a look at the complexity

(Finally) Our First Proof

Decidable Type

A is decidable if $d\colon A + \neg A$.

Key Theorem 1: Finite Codomain

Let $f : A \to B$ a surjective function and A finite. Then B is finite whenever its equality is decidable.

Proof.

- Prop. is shown: get surjective map $g : \mathsf{Fin}_k \to B$.
- By induction on k (k steps). If $k \equiv 0$, then B is empty.
- $k > 0$: decide whether $g(\mathsf{inr}(\star))$ has more than 1 preimage (k values checked, d complexity of one check)
- If not, the induction hypothesis is enough.
- Otherwise, take n yielded by the induction hypothesis on B without $g(\mathsf{inr}(\star))$ and return $n+1$ $T(k) \equiv T(k-1) + kd$

Let's take a look at the complexity

(Finally) Our First Proof

Decidable Type

A is decidable if $d\colon A + \neg A$.

Key Theorem 1: Finite Codomain $\mathcal{O}(|A|^2 d)$

Let $f : A \to B$ a surjective function and A finite. Then B is finite whenever its equality is decidable.

Proof.

- Prop. is shown: get surjective map $g : \mathsf{Fin}_k \to B$.
- By induction on k (k steps). If $k \equiv 0$, then B is empty.
- $k > 0$: decide whether $g(\mathsf{inr}(\star))$ has more than 1 preimage (k values checked, d complexity of one check)
- If not, the induction hypothesis is enough.
- Otherwise, take n yielded by the induction hypothesis on B without $g(\mathsf{inr}(\star))$ and return $n + 1$ $\quad T(k) \equiv T(k-1) + kd$

Let's take a look at the complexity

Type Wars VI: Return of the Definitions

Set Truncation

$\| A \|_0$ is the set of connected components of A. Defined as A quotiented by $\| x = y \|$ for $x, y \colon A$. $| a |_0$ for $a : A$ denotes a quotiented with the relation $\| x = y \|$.

Type Wars VI: Return of the Definitions

Set Truncation

$\| A \|_0$ is the set of connected components of A. Defined as A quotiented by $\| x = y \|$ for $x, y \colon A$. $| a |_0$ for $a : A$ denotes a quotiented with the relation $\| x = y \|$.

Connectedness

A is connected whenever there is an $\omega : \| x = y \|$ for every $x, y : A$

Read: A connected if $\| A \|_0$ has a unique element

- Isomorphic structures are equal
- Hence they are in the same connected component

Slightly more generic:

Homotopy Finiteness

$$\text{is-}\pi_0\text{-finite}(A) :\equiv \text{is-finite} \parallel A \parallel_0$$

$$\text{is-}\pi_{\mathsf{suc}_{\mathbb{N}}(n)}\text{-finite} :\equiv \text{is-finite} \parallel A \parallel_0 \times \prod_{x,y\,:\,A} \text{is-}\pi_n\text{-finite}(x = y)$$

If A is π_0-finite, then it is finite up to isomorphism

Finiteness up to Isomorphism

Key Theorem 2

For B family of π_0-finite types over connected, π_1-finite type A, $\sum_{(x:\,A)} B(x)$ is π_0-finite.

Read: if B family of types finite up to isomorphism over A type with one connected component s.t. its identity types are finite up to isomorphism, then $\sum_{(x:\,A)} B(x)$ is finite up to isomorphism.

Finiteness up to Isomorphism

Key Theorem 2

For B family of π_0-finite types over connected, π_1-finite type A, $\sum_{(x:\,A)} B(x)$ is π_0-finite.

Proof.

- Assume $a : A$, then $f :\equiv b \mapsto (a, b) : B(a) \to \sum_{(x:\,A)} B(x)$ surj.
- $\| f \|_0 : \| B(a) \|_0 \to \left\| \sum_{(x:\,A)} B(x) \right\|_0$ also surj.
- By Key Thm. 1, $\sum_{(x:\,A)} B(x)$ is π_0-finite if it has dec. equality.
- By HoTT shennanigans,

$$(x, y) =_{\left\| \Sigma_{(x:\,A)} B(x) \right\|_0} (x', y') \simeq \left\| \Sigma_{|\, p \,|_0 : \| a = a \|_0} \| \mathrm{tr}_B(p, y) = y' \| \right\|$$

- Then: π_1-finiteness $\Rightarrow \| a = a \|_0$ finite and $\| \mathrm{tr}_B(p, y) = y' \| \simeq |\, \mathrm{tr}_B(p, y) \,|_0 = |\, y' \,|_0$ decidable proposition hence finite.
- Sum of finite types is finite, and a finite prop. is decidable.

□

Let's compute the complexity

Proof.

- Assume $a : A$, then $f :\equiv b \mapsto (a, b) : B(a) \to \sum_{(x:\,A)} B(x)$ surj.
- $\| f \|_0 : \| B(a) \|_0 \to \left\| \sum_{(x:\,A)} B(x) \right\|_0$ also surj.
- By Key Thm. 1, $\sum_{(x:\,A)} B(x)$ is π_0-finite if it has dec. equality.
- By HoTT shennanigans ,

$$(x, y) =_{\left\| \Sigma_{(x:\,A)} B(x) \right\|_0} (x', y') \simeq \left\| \Sigma_{|\,p\,|_0 : \|\, a = a \,\|_0} \| \mathrm{tr}_B(p, y) = y' \| \right\|$$

- Then: π_1-finiteness $\Rightarrow \|\, a = a \,\|_0$ finite and $\| \mathrm{tr}_B(p, y) = y' \| \simeq |\, \mathrm{tr}_B(p, y) \,|_0 = |\, y' \,|_0$ decidable proposition hence finite.
- Sum of finite types is finite, and a finite prop. is decidable.

Just the Once Will Not Hurt

Let's compute the complexity

Proof.

- Assume $a : A$, then $f :\equiv b \mapsto (a, b) : B(a) \to \sum_{(x:\,A)} B(x)$ surj. pretty cheap
- $\| f \|_0 : \| B(a) \|_0 \to \left\| \sum_{(x:\,A)} B(x) \right\|_0$ also surj. pretty cheap
- By Key Thm. 1, $\sum_{(x:\,A)} B(x)$ is π_0-finite if it has dec. equality.
- By HoTT shennanigans (cheap),

$$(x, y) =_{\left\| \Sigma_{(x:\,A)} B(x) \right\|_0} (x', y') \simeq \left\| \Sigma_{|\, p \,|_0 : \|\, a = a \,\|_0} \| \mathrm{tr}_B(p, y) = y' \| \right\|$$

- Then: π_1-finiteness $\Rightarrow \| a = a \|_0$ finite and $\| \mathrm{tr}_B(p, y) = y' \| \simeq | \mathrm{tr}_B(p, y) |_0 = | y' |_0$ decidable proposition hence finite.
- Sum of finite types is finite, and a finite prop. is decidable.

Just the Once Will Not Hurt

Let's compute the complexity

Proof.

- Assume $a : A$, then $f :\equiv b \mapsto (a, b) : B(a) \to \sum_{(x:\ A)} B(x)$ surj. pretty cheap
- $\| f \|_0 : \| B(a) \|_0 \to \left\| \sum_{(x:\ A)} B(x) \right\|_0$ also surj. pretty cheap
- By Key Thm. 1, $\sum_{(x:\ A)} B(x)$ is π_0-finite if it has dec. equality. $\mathcal{O}(|\, \| B(a) \|_0 \,|^2 d)$
- By HoTT shennanigans (cheap),

$$(x, y) =_{\left\| \Sigma_{(x:\ A)} B(x) \right\|_0} (x', y') \simeq \left\| \Sigma_{|\, p \,|_0 : \| a = a \|_0} \| \mathrm{tr}_B(p, y) = y' \| \right\|$$

- Then: π_1-finiteness $\Rightarrow \| a = a \|_0$ finite and $\| \mathrm{tr}_B(p, y) = y' \| \simeq |\, \mathrm{tr}_B(p, y) \,|_0 = |\, y' \,|_0$ decidable proposition hence finite.
- Sum of finite types is finite, and a finite prop. is decidable. (d appears here, $d \equiv \mathcal{O}(|\, \| a = a \|_0 \,|)$ as all op. are needed)

Just the Once Will Not Hurt

Let's compute the complexity $\mathcal{O}(|\,\|\,B(a)\,\|_0\,|^2(|\,\|\,a = a\,\|_0\,|))$

Proof.

- Assume $a : A$, then $f :\equiv b \mapsto (a, b) : B(a) \to \sum_{(x:\,A)} B(x)$ surj. pretty cheap
- $\|\,f\,\|_0 : \|\,B(a)\,\|_0 \to \left\|\,\sum_{(x:\,A)} B(x)\,\right\|_0$ also surj. pretty cheap
- By Key Thm. 1, $\sum_{(x:\,A)} B(x)$ is π_0-finite if it has dec. equality. $\mathcal{O}(|\,\|\,B(a)\,\|_0\,|^2 d)$
- By HoTT shennanigans (cheap),

$$(x, y) =_{\left\|\,\Sigma_{(x:\,A)} B(x)\,\right\|_0} (x', y') \simeq \left\|\,\Sigma_{|\,p\,|_0 : \|\,a=a\,\|_0} \left\|\,\mathrm{tr}_B(p, y) = y'\,\right\|\,\right\|$$

- Then: π_1-finiteness $\Rightarrow \|\,a = a\,\|_0$ finite and $\|\,\mathrm{tr}_B(p, y) = y'\,\| \simeq |\,\mathrm{tr}_B(p, y)\,|_0 = |\,y'\,|_0$ decidable proposition hence finite.
- Sum of finite types is finite, and a finite prop. is decidable. (d appears here, $d \equiv \mathcal{O}(|\,\|\,a = a\,\|_0\,|)$ as all op. are needed)

Quick Application: Finite Semigroups

Complexity of Key Thm. 2: $\mathcal{O}(|\,\|\, B(a) \,\|_0\,|^2(|\,\|\, a = a \,\|_0\,|))$

Finite Semigroup

Finite Type G + associative multiplication $\mu : G \to G \to G$

- Semigroups of order n: $|B(x)| \equiv o(n^{n^2})$
- G is a set (as finite) thus type of assoc. mult. is set
- $\|\, B(x) \,\|_0 \simeq B(x)$ hence same cardinal
- $(G = H) \simeq (G \simeq H)$, $|G \simeq H| \approx \mathcal{O}(n!)$
- G, H sets hence $G = H$ is a set

Total complexity: $\mathcal{O}(n^{2n^2} n!)$

For $n = 2$, 512 operations $\Rightarrow$ some big constant is hidden

Conclusion[1]

What have we seen?

- Over 9000 (lines of λ-terms needed to analyze the proof)
- What is *actually* computed in the proof
- Theoretical complexity: high but others bottlenecks (evaluation, term size)

What's next?

- Better proof complexity-wise: maybe
- Balance between theoretical improvement and term size

[1] Thanks to T. Coquand and J. Höfer for their precious advice

Scribitur ad narrandum, non ad probandum.

Thanks for your attention!

A special thanks to Adrien M. for typesetting the xkcd and his help throughout the internship.