Constructive mathematics in univalent type theory

Martín Hötzel Escardó

University of Birmingham, UK

Summer School Types, Sets and Constructions,
Hausdorff Research Institute for Mathematics, Bonn, May 2018

Classical higher-type computability result

In the model of Kleene–Kreisel spaces, the searchable subspaces are precisely the computable images of the Cantor space $2^{\mathbb{N}}$.

(M.H.E, LMCS'2008.)

(The compact subspaces are precisely the *continuous* images of the Cantor space.)

(So in the computable word, the searchable sets are the effective analogues of the compact spaces in topology.)

Closure under Σ

If X is omniscient/searchable and Y is an X-indexed family of omniscient/searchable types, then so is its disjoint sum $\Sigma(x : X), Y(x)$.

Closure under Π

Not to be expected in general.

E.g. $\mathbb{N}_\infty$ and 2 are omniscient, but in continuous and effective models of type theory, the function space $\mathbb{N}_\infty \to 2$ is not.

In the topological topos, $\mathbb{N}_\infty \to 2$ is a countable discrete space.

Closure under finite products

Theorem (baby Tychonoff) A product of searchable types indexed by a finite type is searchable.

We will need this form of closure under Π

Theorem (micro Tychonoff)
A product of searchable types indexed by a subsingleton type is itself searchable.

That is, if X is a subsingleton, and Y is an X-indexed family of searchable types, then the type $\Pi(x : X), Y(x)$ is searchable.

We will need this form of closure under Π

Theorem (micro Tychonoff)
A product of searchable types indexed by a subsingleton type is itself searchable.

That is, if X is a subsingleton, and Y is an X-indexed family of searchable types, then the type $\Pi(x : X), Y(x)$ is searchable.

This cannot be proved if searchability is replaced by omniscience (that is, if we don't assume that every $Y(x)$ is pointed).

This is easy with excluded middle, but we are not assuming it.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \dots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to (X \to 0)$.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to (X \to 0)$.
6. $(X \to 0) \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to (X \to 0)$.
6. $(X \to 0) \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), (X \to 0) \to p(z) = 1$.

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to (X \to 0)$.
6. $(X \to 0) \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), (X \to 0) \to p(z) = 1$.
7. By transitivity of $\to$, we get

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to (X \to 0)$.
6. $(X \to 0) \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), (X \to 0) \to p(z) = 1$.
7. By transitivity of $\to$, we get

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to p(z) = 1$, so

Theorem A subsingleton-indexed product of searchable types is searchable.

1. Let X subsingleton, $Y(x)$ searchable for every $x : X$.
2. $Z \stackrel{\text{def}}{=} \Pi(x : X), Y(x)$.

   We have $\Pi(x : X), (Z \simeq Y(x))$ and $(X \to 0) \to (Z \simeq 1)$.
3. Let $p : Z \to 2$.
4. Construct $z_0(x) \stackrel{\text{def}}{=} \ldots$ in Z using the first equivalence.
5. $X \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), X \to p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to (X \to 0)$.
6. $(X \to 0) \to p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$.

   $p(z_0) = 1 \to \Pi(z : Z), (X \to 0) \to p(z) = 1$.
7. By transitivity of $\to$, we get

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 0 \to p(z) = 1$, so

   $p(z_0) = 1 \to \Pi(z : Z), p(z) = 1$. Q.E.D.

Amusing consequence, tangential to our development

Consider the truncated version of LPO, which is logically equivalent to the original, untruncated one.

Corollary. *The type* $\mathbb{N}^{LPO}$ *is searchable.*

- The reason is that LPO implies that $\mathbb{N}$ is searchable, and so this is a product of searchable types.

  Even though the searchability of $\mathbb{N}$ is undecided!

- If LPO holds, the type of the corollary is $\mathbb{N}$.
- If LPO fails, it is the contractible type 1.
- As LPO is undecided, we don't know what the type $\mathbb{N}^{\mathsf{LPO}}$ "really is".
- Whatever it is, however, it is always searchable.

Disjoint sum with a point at infinity

Theorem
The disjoint sum of a countable family of searchable sets with a point at infinity is searchable.

We need to say how we add a point at infinity.

The type $1 + \Sigma(n : \mathbb{N}), X(n)$ won't do, of course.

We will do this in a couple of steps.

Injectivity of the universe of types

Theorem
For any embedding $e : A \to B$, every $X : A \to U$ extends to some $Y : B \to U$ along e, up to equivalence,
$$\Pi(a : A), (Y(e(a)) \simeq X(a)).$$

A map $e : A \to B$ is called an embedding iff its fibers $e^{-1}(b)$,

$$\Sigma(a : A), f(a) = b,$$

are all subsingletons.

Injectivity of the universe of types

Theorem

For any embedding $e : A \to B$, every $X : A \to U$ extends to some $Y : B \to U$ along e, up to equivalence.

Two constructions:

1. We have the "maximal" extension $Y = X/e$.

$$
\begin{aligned}
(X/e)(b) &= \Pi \left(s : e^{-1}(b)\right), X(\mathrm{pr}_1\, s) \\
&\simeq \Pi(a : A),\ e(a) = b \ \to\ X(a).
\end{aligned}
$$

Injectivity of the universe of types

Theorem

For any embedding $e : A \to B$, every $X : A \to U$ extends to some $Y : B \to U$ along e, up to equivalence.

Two constructions:

1. We have the "maximal" extension $Y = X/e$.

$$\begin{aligned}(X/e)(b) &= \Pi\left(s : e^{-1}(b)\right), X(\mathrm{pr}_1\, s) \\ &\simeq \Pi(a : A),\; e(a) = b \;\to\; X(a).\end{aligned}$$

2. And also the "minimal" extension $Y = X \setminus e$.

$$\begin{aligned}(X \setminus e)(b) &= \Sigma\left(s : e^{-1}(b)\right), X(\mathrm{pr}_1\, s) \\ &\simeq \Sigma(a : A),\; (e(a) = b) \;\times\; X(a).\end{aligned}$$

The first one works our purposes.

Injectivity of the universe of types

Let $e : A \to B$ be an embedding and $X : A \to U$.

Consider the extended type family $X/e : B \to U$ defined above:

$$(X \setminus e)(b) = \Pi \left(s : e^{-1}(b)\right), X(\mathrm{pr}_1\, s)$$

We have

1. For all $b : B$ *not* in the image of the embedding,

$$(X/e)(b) \simeq 1.$$

2. If for all $a : A$, the type $X(a)$ is searchable too, then for all $b : B$ the type $(X/e)(b)$ is searchable, by micro-Tychonoff.

3. Hence if additionally B is searchable, the type $\Sigma(b : B), (X/e)(b)$ is searchable too.

4. We are interested in $A = \mathbb{N}$ and $B = \mathbb{N}_\infty$, which gives the disjoint sum of $X(a)$ with a point at infinity.

A map $L : (\mathbb{N} \to U) \to U$

Let $e : \mathbb{N} \to \mathbb{N}_\infty$ be the natural embedding.

Given $X : \mathbb{N} \to U$, first take $X/e : \mathbb{N}_\infty \to U$

This step adds a point at infinity to the sequence.

We then sum over $\mathbb{N}_\infty$, to get $L(X)$:

$$L(X) = \Sigma(u : \mathbb{N}_\infty), (X/e)(u).$$

By micro-Tychonoff, L maps any sequence of searchable types to a searchable type.

Iterating this map $L : (\mathbb{N} \to U) \to U$

We get (very large!) searchable ordinals, with the property that any inhabited *decidable* subset has a least element.

Iterating this map $L : (\mathbb{N} \to U) \to U$

We get (very large!) searchable ordinals, with the property that any inhabited *decidable* subset has a least element.

They are all countable.

Or rather they each have a countable subset with empty complement.

Iterating this map $L : (\mathbb{N} \to U) \to U$

We get (very large!) searchable ordinals, with the property that any inhabited *decidable* subset has a least element.

They are all countable.

Or rather they each have a countable subset with empty complement.

An ordinal is a type X with a transitive, extensional, relation satistfying well-foundeness for decidable subsets.

A functor $F : U \to U$

$F(X) = L(\lambda n.X)$, which is equivalent to $\Sigma(u : \mathbb{N}_\infty), \Pi(n : \mathbb{N}), X^{e(n)=u}$.

An equivalent coinductive definition of F is given by constructors

$$\begin{aligned} \mathrm{zero} &: X \to F(X), \\ \mathrm{succ} &: F(X) \to F(X). \end{aligned}$$

This is the so-called delay monad.

Curry–Howard "excluded middle"

Theorem of MLTT$+\|-\|$. The following are logically equivalent:

1. $\Pi(X:\mathcal{U}), X + \neg X$.
2. $\Pi(X:\mathcal{U}), \neg\neg X \to X$.
3. $\Pi(X:\mathcal{U}), \|X\| \to X$.
4. $\Pi(X:\mathcal{U}), \Sigma(f : X \to X), \Pi(x, y : X), f(x) = f(y)$.

- This is more like global choice than excluded middle.

  We can pick a point of every non-empty type.
- It implies that all types are sets, making univalent type theory trivial.
- False in the presence of univalence.

  Not possible to choose a point of every set in an invariant-under-isomorphism way.

Univalent excluded middle

The following are equivalent:

1. $\Pi(P : \mathcal{U}), \mathrm{isProp}(P) \to P + \neg P$,
2. $\Pi(P : \mathcal{U}), \mathrm{isProp}(P) \to \neg\neg P \to P$,
3. $\Pi(X : \mathcal{U}), \neg\neg X \to \|X\|$.

Which is consistent with univalent type theory.

Correct formulation of unique existence

- Not $(\Sigma(x : X), A(x)) \times (\Pi(x, y : X), A(x) \times A(y) \to x = y)$.
- Instead $\mathrm{isSingleton}(\Sigma(x : X), A(x))$.

  Especially when formulating universal properties.
- A unique $x : X$ such that $A(x)$ is not enough.
- What is really needed is a unique *pair* (x, a) with $x : X$ and $a : A(x)$.

  Like in category theory again.

  Unless all types are sets.

Correct formulation of “at most one”

$$\mathrm{isProp}(\Sigma(x : X), A(x)).$$

Choice just holds in Curry–Howard logic

Let $X, Y : \mathcal{U}$ be types and $R : X \times Y \to \mathcal{U}$ be a relation.

$$(\Pi(x : X), \Sigma(y : Y), R(x, y)) \to \Sigma(f : X \to Y), \Pi(x : X), R(x, f(x)).$$

Moreover, the implication is a type equivalence.

However, univalent choice implies univalent excluded middle

$$(\Pi(x : X), \|\Sigma(y : Y(x)), R(x,y)\|) \to \|\Sigma(f : \Pi(x : X), Y(x)), \Pi(x : X), R(x, f(x))\|.$$

The assumptions are that $\mathrm{isSet}\, X$ and $\mathrm{isSet}\, Y(x)$ for all $x : X$, and we are given $R : (\Sigma(x : X), Y(x)) \to \Omega$.

This form of choice is consistent with univalent type theory.

Equivalent formulation of univalent choice in pure MLTT (and hence in UTT)

$$(\Pi(x : X), \neg\neg\Sigma(y : Y(x)), R(x, y)) \to \neg\neg\Sigma(f : \Pi(x : X), Y(x)), \Pi(x : X), R(x, f(x)).$$

This is because if we have excluded middle then the propositional truncation is double negation, but choice (expressed with double negation) gives excluded middle.

But we do get unique choice

"Moral"

1. Σ is used to express given $\boxed{\text{structure}}$ or $\boxed{\text{data}}$ in general.

   Cf. the type of groups.

2. Truncated Σ is used to express $\boxed{\text{existence}}$. (Still constructive.)

   - But, even better, in practice, one is encouraged to use Σ so that it produces univalent propositions without the need of truncation (if we can).
   - A crucial example is Voevodsky's primary notion of equivalence.

     (But we have seen additional examples.)

3. At the moment, it seems to be an art to decide whether particular mathematical statements should be formulated as giving structure/data or propositions.

4. But the main point is that $\boxed{\text{this mathematical language allows the distinction.}}$