Infinite, exhaustibly searchable sets
in dependent type theory and everywhere

Martín Hötzel Escardó

University of Birmingham, UK

CCC'2015, Kochel, Germany

Continuity, Computability and Constructivity

We don't assume continuity axioms.

Computability and Constructivity

We don’t need to talk about computability.

Constructivity

Constructivity is implicit.
Compatible with classical mathematics.

“Absolute” mathematics

Rendered in type theory here.

We will prove instances of excluded middle.

Mathematics in dependent type theory

I'll work in intensional Martin-Löf type theory (MLTT).

I'll explain it by example, and I will work informally.

But I have also written the formal versions of the proofs in Agda notation.

Two benefits:

1. Proofs are programs, and hence we can run them.
2. Agda checks the proofs.

However, what we do here (or more generally in any piece of mathematics developed in MLTT) doesn't depend on the fact that we can run our proofs.

Here is how we can formulate LPO in MLTT

$$\Pi(p : \mathbb{N} \to 2).(\Sigma(n : \mathbb{N}).p(n) = 0) + (\Pi(n : \mathbb{N}).p(n) = 1)$$

This is a type, built as follows.

- $\mathbb{N}$ is the type of natural numbers.
- 2 is the type of binary numbers $0, 1$.
- $\mathbb{N} \to 2$ is the type of functions from $\mathbb{N}$ to 2. We also write $2^{\mathbb{N}}$.
- Given a family of types A indexed by the type $\mathbb{N} \to 2$, the product type $\Pi(p : \mathbb{N} \to 2).A(p)$ consists of the functions that map a given $p : \mathbb{N} \to 2$ to an element of $A(p)$.
- Given a family of types B indexed by the type $\mathbb{N}$, its disjoint union $\Sigma(n : \mathbb{N}).B(n)$ consists of the pairs (n, b) with $n : \mathbb{N}$ and $b : B(n)$.
- Given $m, n : 2$, the equality type $m = n$, has precisely one element when m and n are the same, and is empty otherwise. (This can be defined by four cases.)
- Given two types X and Y, the type $X + Y$ is their disjoint union.

An equivalent formulation is this

$$\Pi(p : \mathbb{N} \to 2).(\Sigma(n : \mathbb{N}).p(n) = 0) + (\neg\Sigma(n : \mathbb{N}).p(n) = 0)$$

Here

- $\neg X$ abbreviates $X \to 0$.
- 0 is the empty type.
- Later we will need the one-point type 1.
- This is all about theory that we need to know for our purposes.

LPO is undecided

$$\Pi(p : \mathbb{N} \to 2).(\Sigma(n : \mathbb{N}).p(n) = 0) + (\neg\Sigma(n : \mathbb{N}).p(n) = 0)$$

1. In type theory we don't prove propositions.
2. We instead inhabit types (and still say we prove propositions).
3. A meta-theorem is that MLTT doesn't inhabit LPO or ¬LPO.
4. Each of them is consistent with MLTT.
5. LPO is undecided, and we'll keep it that way.
6. But will say it is a taboo.

We now make $\mathbb{N}$ larger by adding a point at infinity

Let $\mathbb{N}_\infty$ be the type of decreasing binary sequences.

$$\mathbb{N}_\infty \stackrel{\text{def}}{=} \Sigma(\alpha : 2^{\mathbb{N}}).\Pi(n : \mathbb{N}).\alpha(n) = 0 \to \alpha(n+1) = 0.$$

1. The type $\mathbb{N}$ embeds into $\mathbb{N}_\infty$ by mapping the number $n : \mathbb{N}$ to the sequence $\underline{n} \stackrel{\text{def}}{=} 1^n 0^\omega$.
2. A point not in the image of this is $\infty \stackrel{\text{def}}{=} 1^\omega$.

Theorem

$$\Pi(p : \mathbb{N}_\infty \to 2).(\Sigma(x : \mathbb{N}_\infty).p(x) = 0) + (\neg\Sigma(x : \mathbb{N}_\infty).p(n) = 0)$$

1. This is LPO with $\mathbb{N}$ replaced by $\mathbb{N}_\infty$.
2. We don't use continuity, which anyway is not available in MLTT.
3. This theorem actually makes sense in any variety of constructive mathematics (JSL 2013).

WLPO is also undecided by MLTT

$$\Pi(p : \mathbb{N} \to 2).(\Pi(n : \mathbb{N}).p(n) = 1) + (\neg\Pi(x : \mathbb{N}).p(n) = 1)$$

But we have:

Theorem $\Pi(p : \mathbb{N}_\infty \to 2).(\Pi(n : \mathbb{N}).p(\underline{n}) = 1) + (\neg\Pi(n : \mathbb{N}).p(\underline{n}) = 1)$

1. The point is that now we quantify over $\mathbb{N}$, although the function p is defined on $\mathbb{N}_\infty$.
2. This again holds in any variety of constructive mathematics and doesn't rely on continuity axioms (JSL'2013).

Some consequences

1. Every function $f : \mathbb{N}_\infty \to \mathbb{N}$ is constant or not.
2. Any two functions $f, g : \mathbb{N}_\infty \to \mathbb{N}$ are equal or not.
3. Any function $f : \mathbb{N}_\infty \to \mathbb{N}$ has a minimum value, and it is possible to find the point at which the minimum value is attained.
4. For any function $f : \mathbb{N}_\infty \to \mathbb{N}$ there is a point $x : \mathbb{N}_\infty$ such that if f has a maximum value, the maximum value is x.
5. Any function $f : \mathbb{N}_\infty \to \mathbb{N}$ is not continuous, or not-not continuous.
6. There is a non-continuous function $f : \mathbb{N}_\infty \to \mathbb{N}$ iff WLPO holds.

Are there more types like $\mathbb{N}_\infty$?

1. Plenty.
2. Our business here is how to construct them.
3. But we pause to reflect first.

What have we been doing?

Giving examples of types X and properties P of X such that the assertion

for all $x : X$, either $P(x)$ or not $P(x)$

just holds.

1. In classical mathematics, we assume excluded middle.
2. Here we investigate how much of it just holds.

   No philosophy or meta-mathematics, except for side-remarks.

   We just prove mathematical theorems.

Two notions

Definition (Omniscient type) A type X is omniscient if for every $p : X \to 2$, the assertion that $p(x) = 0$ for some $x : X$ is decidable.

In symbols:

$$\Pi(p : X \to 2).(\Sigma(x : X).p(x) = 0) + (\neg\Sigma(x : X).p(x) = 0).$$

Definition (Searchable type) A type X is searchable if for every $p : X \to 2$ we can find $x_0 : X$, called a universal witness for p, such that if $p(x_0) = 1$, then $p(x) = 1$ for all $x : X$.

In symbols,

$$\Pi(p : X \to 2).\Sigma(x_0 : X).p(x_0) = 1 \to \Pi(x : X).p(x) = 1.$$

These are sort of notions of compactness without mentioning topology or assuming continuity principles.

Their relationship

Definition
$\text{omniscient}(X) \stackrel{\text{def}}{=} \Pi(p : X \to 2).(\Sigma(x : X).p(x) = 0) + (\neg\Sigma(x : X).p(x) = 0).$

Definition
$\text{searchable}(X) \stackrel{\text{def}}{=} \Pi(p : X \to 2).\Sigma(x_0 : X).p(x_0) = 1 \to \Pi(x : X).p(x) = 1.$

Proposition A type X is searchable iff it is inhabited and omniscient:

$$\text{searchable}(X) \iff X \times \text{omniscient}(X).$$

A few theorems rely on inhabitation, using the notion of searchability.

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

Limited form of closure under Π

Definition A type X is a subsingleton if all its elements are equal:

$$\Pi(x, y : X).x = y$$

Theorem A product of searchable types indexed by a subsingleton type is 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.

I don't think this this can be proved if searchability is replaced by omniscience (that is, if we don't assume that every $Y(x)$ is inhabited).

This is easy with excluded middle, but we are not assuming it.
(This is a kind of micro Tychonoff Theorem.)

Corollary A product of searchable types indexed by a subtype of a finite type is searchable.

Proof

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

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

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 Given any A-indexed family of types X, and given any type B having A embedded as a subtype via $e : A \to B$, we can extend X to a B-indexed family Y of types:

$$\Pi(a : A).Y(e(a)) \cong X(a).$$

Moreover, this can be done so that

1. for all $b : B$ not in the image of the embedding $e : A \to B$, we have $Y(a) \cong 1$,
2. If for all $a : A$ the type $X(a)$ is searchable, then for all $b : B$ the type $Y(b)$ is searchable.

In particular, under these assumptions, the type $\Sigma(b : B).Y(b)$ is searchable if B is searchable. This relies on the micro-Tychonoff theorem.

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.

Interating this

Gives searchable ordinals.

Unfortunately, no time to discuss this.

If there is any time left, I will sketch some selected proofs

Take away conclusion

Plenty of instances of excluded middle, of the form

for all $x : X$*, either* $P(x)$ *or not* $P(x)$*,*

with X infinite, just hold.

This is so in all varieties of constructive mathematics (and in classical mathematics without using excluded middle to prove them).

Computability or continuity considerations are not needed.

I am not sure whether it was a good idea to present this in type theory in this talk.

But I did formalize it in type theory, in Agda notation, to illustrate that computability considerations are not needed to run our theorems.

And to convince myself that type theory is a workable foundation of (constructive or non-constructive) mathematics.