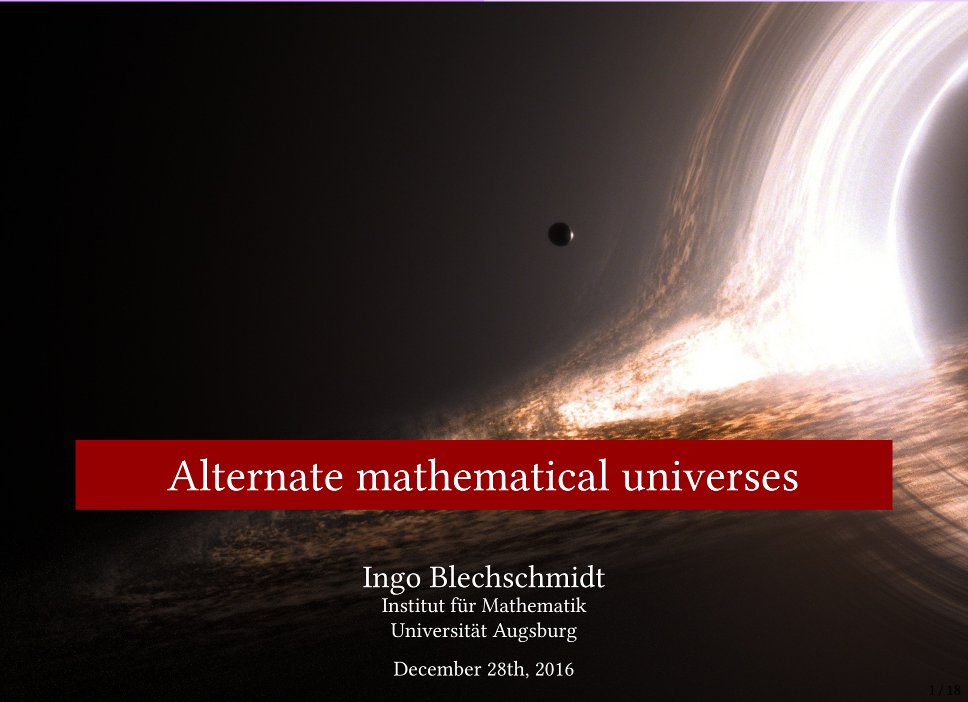


Alternate mathematical universes

Ingo Blechschmidt
Institut für Mathematik
Universität Augsburg

December 28th, 2016

On truth

"There are infinitely many prime numbers."

"There are exactly five platonic solids."

"There are more real numbers than integers."

- Mathematical statements are thought to be absolutely objective and independent of human sentiments.

On truth

"There are infinitely many prime numbers."

"There are exactly five platonic solids."

"There are more real numbers than integers."

- Mathematical statements are thought to be absolutely objective and independent of human sentiments.
- But in fact, their truth depends on the **choice of mathematical universe** ("topos").
- The usual axioms of logic hold in the **standard topos** Set.
- In other topoi, slightly different axioms are true.

Examples

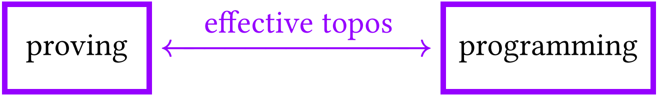
There are topoi in which …

- there is an intermediate infinity between $\mathbb{N}$ and $\mathbb{R}$,
- there is no such intermediate infinity,
- there are **infinitesimal real numbers** ε such that $\varepsilon \neq 0$ but $\varepsilon^2 = 0$, or
- any function $\mathbb{R} \to \mathbb{R}$ is a polynomial.

Examples

There are topoi in which …

- there is an intermediate infinity between $\mathbb{N}$ and $\mathbb{R}$,
- there is no such intermediate infinity,
- there are **infinitesimal real numbers** ε such that $\varepsilon \neq 0$ but $\varepsilon^2 = 0$, or
- any function $\mathbb{R} \to \mathbb{R}$ is a polynomial.



The effective topos

- "$1 + 1 = 2$."
  True in Set, true in $\mathrm{Eff}(\mathrm{TM})$.

- "Any number is either prime or not."
  Trivially true in Set, nontrivially true in $\mathrm{Eff}(\mathrm{TM})$.

- "Any function $\mathbb{N} \to \mathbb{N}$ is either the zero function or not."
  Trivially true in Set, false in $\mathrm{Eff}(\mathrm{TM})$.

- "Any function $\mathbb{N} \to \mathbb{N}$ is computable by a Turing machine."
  False in Set, trivially true in $\mathrm{Eff}(\mathrm{TM})$.

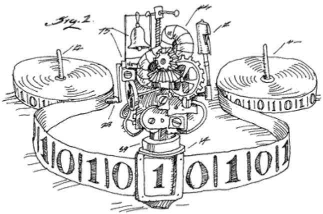
- "Any function $\mathbb{R} \to \mathbb{R}$ is continuous."
  False in Set, true in $\mathrm{Eff}(\mathrm{TM})$.

The effective topos

There are many **models of computation**:

- Turing machines
- Lambda calculus
- Perl programs (running on idealized computers)
- Computers in the real physical world

For any such model $\mathcal{M}$, there is a special topos $\mathrm{Eff}(\mathcal{M})$, the associated **effective topos**.



The effective topos

There are many **models of computation**:

- Turing machines
- Lambda calculus
- Perl programs (running on idealized computers)
- Computers in the real physical world

For any such model $\mathcal{M}$, there is a special topos $\mathrm{Eff}(\mathcal{M})$, the associated **effective topos**.

$\mathrm{Eff}(\mathcal{M}) \models$ "For any number n there is a prime $p > n$."

means:

There is a program which reads a number n as input and outputs a prime number $p > n$.

Our first steps in the effective topos

$\mathrm{Eff}(\mathcal{M}) \models$ "For any number n there is a prime $p > n$."

means:

There is a program which reads a number n as input and outputs a prime number $p > n$.

Our first steps in the effective topos

$\mathrm{Eff}(\mathcal{M}) \models$ "For any number n there is a prime $p > n$."
means:
There is a program which reads a number n as input and outputs a prime number $p > n$.

$\mathrm{Eff}(\mathcal{M}) \models$ "Any number has a prime factor decomposition."
means:
There is a program which reads a number n as input and outputs a list of primes, the product of which is n.

Our first steps in the effective topos

$\mathrm{Eff}(\mathcal{M}) \models$ "For any number n there is a prime $p > n$."
means:
There is a program which reads a number n as input and outputs a prime number $p > n$.

$\mathrm{Eff}(\mathcal{M}) \models$ "Any number has a prime factor decomposition."
means:
There is a program which reads a number n as input and outputs a list of primes, the product of which is n.

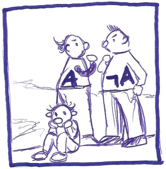
$\mathrm{Eff}(\mathcal{M}) \models$ "Any number is either prime or not prime."
means:
There is a program which reads a number n as input and outputs YES or NO depending on whether n is prime or not.

What's true in alternate topoi?

Metatheorem: If a statement has a **constructive proof**, then it holds in **any topos**.

Constructive logic is like classical logic, except we don't suppose the **law of excluded middle** (LEM), which says:

- "Any statement is either true or not true."
- "If a statement is *not not* true, then it's true."



Nonconstructive proofs

A number is **rational** if and only if its decimal digits eventually repeat.

- $\frac{21}{13}$ and 37 are rational.
- $\sqrt{2}$ and π are irrational.

Nonconstructive proofs

A number is **rational** if and only if its decimal digits eventually repeat.

- $\frac{21}{13}$ and 37 are rational.
- $\sqrt{2}$ and π are irrational.

Theorem. There are **irrational** numbers x und y such that x^y is rational.

Nonconstructive proofs

A number is **rational** if and only if its decimal digits eventually repeat.

- $\frac{21}{13}$ and 37 are rational.
- $\sqrt{2}$ and π are irrational.

Theorem. There are **irrational** numbers x und y such that x^y is rational.

Proof. Either $\sqrt{2}^{\sqrt{2}}$ is rational or not.

1. In the first case we are done.
2. In the second case we set $x := \sqrt{2}^{\sqrt{2}}$ and $y := \sqrt{2}$. Then $x^y = \sqrt{2}^{\sqrt{2}\cdot\sqrt{2}} = \sqrt{2}^2 = 2$ is rational.

Appreciating constructive logic

At first sight, dropping the axiom of excluded middle looks like a sad thing to do. It's a useful axiom! However:

- The axiom is not needed as often as one would think.
- The abstinence is good for your mental hygiene.
- Constructive logic allows for finer distinctions.
- From constructive proofs one can mechanically extract programs which witness the proved statements.
- **Dropping the law of excluded middle allows to add curious unconvential axioms.**

LEM for equality of functions

$\mathrm{Eff}(\mathrm{TM}) \models$ "Any function $f : \mathbb{N} \to \mathbb{N}$ is either the zero function or not."

means:

There is a program which reads the source of a program M, which calculates a function $\mathbb{N} \to \mathbb{N}$, as input, and finds out whether M always yields zero or not.

That's false.

LEM for equality of functions

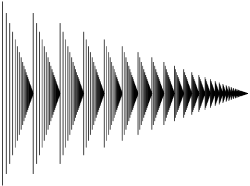
Eff(TM) $\models$ "Any function $f : \mathbb{N} \to \mathbb{N}$ is either the zero function or not."

means:

There is a program which reads the source of a program M, which calculates a function $\mathbb{N} \to \mathbb{N}$, as input, and finds out whether M always yields zero or not.

That's false.

The statement is true in Eff(STM), the effective topos associated to **super Turing machines**.



LEM for the halting problem

$\mathrm{Eff}(\mathrm{TM}) \models$ "Any program halts or doesn't halt."

means:

There is a program which reads the source of a program M as input and finds out whether M halts or not.

That's false.

LEM for the halting problem

$\mathrm{Eff}(\mathrm{TM}) \models$ "Any program halts or doesn't halt."

means:

There is a program which reads the source of a program M as input and finds out whether M halts or not.

That's false.

The statement is true in $\mathrm{Eff}(\mathrm{STM})$.

LEM for equality of real numbers

The statement

> "Every real number is either zero or not zero."

is in constructive logic equivalent to

> "Every Turing machine halts or doesn't halt.",

so it's not true in $\mathrm{Eff}(\mathrm{TM})$, but in $\mathrm{Eff}(\mathrm{STM})$.

LEM for equality of real numbers

The statement

> "Every real number is either zero or not zero."

is in constructive logic equivalent to

> "Every Turing machine halts or doesn't halt.",

so it's not true in Eff(TM), but in Eff(STM).

For a Turing machine M consider the real number $0.000\ldots$, whose n'th decimal digit is a one if and only if M halts after step n.

For a real number x consider the Turing machine which searches the digits of x for a nonzero digit.

Markov's principle

$\mathrm{Eff}(\mathrm{TM}) \models$ "For any function $f : \mathbb{N} \to \mathbb{N}$ which is not the zero function, there is a number $n \in \mathbb{N}$ such that $f(n) \neq 0$."

means:

There is a program which reads the source of a program M, which calculates a function $\mathbb{N} \to \mathbb{N}$ which is not the zero function, as input and outputs a number n such that $M(n)$ is not zero.

Markov's principle

$\mathrm{Eff}(\mathrm{TM}) \models$ "For any function $f : \mathbb{N} \to \mathbb{N}$ which is not the zero function, there is a number $n \in \mathbb{N}$ such that $f(n) \neq 0$."

means:

There is a program which reads the source of a program M, which calculates a function $\mathbb{N} \to \mathbb{N}$ which is not the zero function, as input and outputs a number n such that $M(n)$ is not zero.

That's true! By unbounded search.

Searching uncountable sets

> "For any function $f : \mathbb{N} \to \mathbb{N}$ from numbers to numbers, there either exists a number n such that $f(n)$ is one or there is no such number."

This statement is false in $\mathrm{Eff}(\mathrm{TM})$.

> "For any function $P : L(\mathbb{B}) \to \mathbb{B}$ from infinite lists of booleans to booleans, there either exists a list x such that $P(x)$ is true or there is no such list."

Searching uncountable sets

> "For any function $f : \mathbb{N} \to \mathbb{N}$ from numbers to numbers, there either exists a number n such that $f(n)$ is one or there is no such number."

This statement is false in $\mathrm{Eff}(\mathrm{TM})$.

> "For any function $P : L(\mathbb{B}) \to \mathbb{B}$ from infinite lists of booleans to booleans, there either exists a list x such that $P(x)$ is true or there is no such list."

This statement is true in $\mathrm{Eff}(\mathrm{TM})$!

The Church–Turing thesis

The **Church–Turing thesis** states: If a function $f : \mathbb{N} \to \mathbb{N}$ is calculable in the real world, then it's also calculable by a program (running on a Turing machine).

$\mathrm{Eff}(\mathrm{TM}) \models$ "Any function $f : \mathbb{N} \to \mathbb{N}$ is calculable by a program."

means:

There is a program which reads the source of a program calculating a function $f : \mathbb{N} \to \mathbb{N}$ as input and outputs the source of a program which calculates f.

That's trivial! "cat" is the looked-for program.

The Church–Turing thesis

The **Church–Turing thesis** states: If a function $f : \mathbb{N} \to \mathbb{N}$ is calculable in the real world, then it's also calculable by a program (running on a Turing machine).

$\mathrm{Eff}(\mathrm{TM}) \models$ "Any function $f : \mathbb{N} \to \mathbb{N}$ is calculable by a program."

means:

There is a program which reads the source of a program calculating a function $f : \mathbb{N} \to \mathbb{N}$ as input and outputs the source of a program which calculates f.

That's trivial! "cat" is the looked-for program.

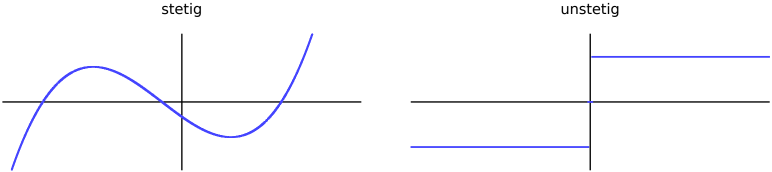
In $\mathrm{Eff}(\mathrm{STM})$ and $\mathrm{Eff}(\lambda\mathrm{C})$ the statement is false.

Automatic continuity

The following statement is wildly **false** in Set:

“Every function $f : \mathbb{R} \to \mathbb{R}$ is continuous.”

A function f is **continuous** if and only if, for calculating $f(x)$ to finitely many digits, finitely many digits of x suffice.

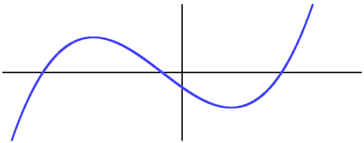


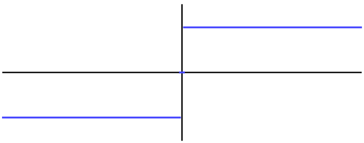
Automatic continuity

The following statement is wildly **false** in Set:

"Every function $f : \mathbb{R} \to \mathbb{R}$ is continuous."

A function f is **continuous** if and only if, for calculating $f(x)$ to finitely many digits, finitely many digits of x suffice.





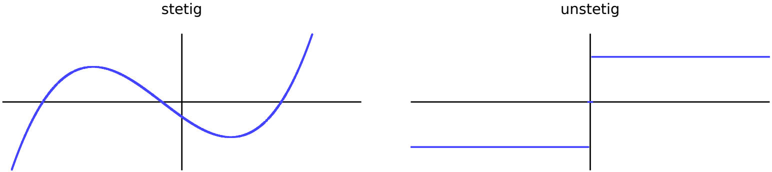
True in Eff(TM).

Automatic continuity

The following statement is wildly **false** in Set:

"Every function $f : \mathbb{R} \to \mathbb{R}$ is continuous."

A function f is **continuous** if and only if, for calculating $f(x)$ to finitely many digits, finitely many digits of x suffice.



True in Eff(TM). True in Eff(RW), if private communication channels are possible and only finitely many computational steps can be executed in finite time.

Wrapping up

- Effective topoi are a good vehicle for studying the nature of computation.
- The effective topoi build links between constructive mathematics and programming.
- Topoi allow for curious dream axioms.
- Topoi have a geometric flavour: points, subtopoi, maps between topoi.

Wrapping up

- Effective topoi are a good vehicle for studying the nature of computation.
- The effective topoi build links between constructive mathematics and programming.
- Topoi allow for curious dream axioms.
- Topoi have a geometric flavour: points, subtopoi, maps between topoi.

There is more to mathematics than the standard topos.