Logipedia:
towards a Wikipedia of formal proofs

I. What is a formal proof?

What is a proof?

For 5000 years: algorithms: find the value of $98 + 99$

But, 2500 years ago: find n and p such that $n^2 = 2 \times p^2$

Try

$n = 5,\ p = 2\colon\ 5^2 = 25 \neq 8 = 2 \times 2^2$

$n = 14,\ p = 10\colon\ 14^2 = 196 \neq 200 = 2 \times 10^2$

$n = 7,\ p = 5\colon\ 7^2 = 49 \neq 50 = 2 \times 5^2$

...

Yes, but for how long?

A new way to answer a question

1. If there is a solution n, p, then $n^2 = 2 \times p^2$ even, thus n even
Useless to try $n = 5$, $p = 2$

2. As $n = 2 \times m$, $4 \times m^2 = 2 \times p^2$ thus $p^2 = 2 \times m^2$ even, thus p even

3. As p also even $p = 2 \times q$, thus $m^2 = 2 \times q^2$ is a smaller solution

4. Thus, if there is a solution n, p, then there is a smaller one, then another smaller one, then another smaller one... Impossible

A new way to answer a question

1. If there is a solution n, p, then $n^2 = 2 \times p^2$ even, thus n even
Useless to try $n = 5$, $p = 2$

2. As $n = 2 \times m$, $4 \times m^2 = 2 \times p^2$ thus $p^2 = 2 \times m^2$ even, thus p even

3. As p also even $p = 2 \times q$, thus $m^2 = 2 \times q^2$ is a smaller solution

4. Thus, if there is a solution n, p, then there is a smaller one, then another smaller one, then another smaller one... Impossible

A contrast with multiplication

Progress step by step

With many possibilities at each step
A high probability to get stuck
Little hope to solve the problem

Rules to build proofs

$$\frac{A \Rightarrow B \qquad A}{B}$$

$$\frac{A \wedge B}{A}$$

$$\frac{A \wedge B}{B}$$

...

Ten to twenty rules enough to build all proofs

Frege, Hilbert, Gentzen... after Aristotle, Leibniz...

Undecidability, decidability

Church and Turing (1936): no algorithm to decide if a proposition has a proof or not

But also... an algorithm to decide if a proof is correct or not

$$\dfrac{\dfrac{(P \Rightarrow Q) \wedge P}{P \Rightarrow Q} \qquad \dfrac{(P \Rightarrow Q) \wedge P}{P}}{Q}$$

$$\dfrac{\dfrac{(Q \Rightarrow P) \wedge P}{Q \Rightarrow P} \qquad \dfrac{(Q \Rightarrow P) \wedge P}{P}}{Q}$$

Thirty years later

1936: an algorithm to decide if a proof is correct or not

1967: a program to decide if a proof is correct or not
AUTOMATH: De Bruijn

A formal proof: a proof sufficiently detailed to be checked by a program
~~then there is a smaller one, then another smaller one, then another smaller one... Impossible~~

What for?

- From 1746 to 1815, proofs of the fundamental theorem of algebra: d'Alembert, Euler, Foncenex, Lagrange, Laplace, Gauss, Argand
  All these proofs relied on claims which lacked adequate justification

- Go with the evolution of the notion of proof: Hales' theorem proved (1998) using a computer algebra system, a formal proof (2014)

- Program correctness (transportation, health, energy...)

Which axioms for mathematics?

An answer: Zermelo's set theory (1908)

Permits to axiomatize mathematics in principle but not in facts
(not a good notion of function, not a good notion of computation)

Other axioms: type theory (ies)

Multiplication of languages for mathematics (like programming languages)

II. Dedukti and Logipedia

The importance of formats

In the good old time, while developing a text processing system, we decided

- to code "a" with 97 and "b" with 98
- or the other way around

Implicitly defined a format

Today: we define first a format (example: HTML) and software must comply with the format

An idea that has (not yet) reached the domain of formal proofs

"A Coq proof of the four color theorem"

Interoperability, durability...

Reverse mathematics

Why is it more difficult for formal proofs?

Because we cannot go too far

Euclidean geometry $\not\leftrightarrow$ Riemannian geometry

$\mathrm{ZF} \not\leftrightarrow \mathrm{ZFC}$

But...

A ZF proof can be "translated" to ZFC

A ZFC proof that does not use the axiom of choice can be "translated" to ZF

Transform proofs by hand or automatically to eliminate the axiom of choice, the excluded middle...

There exists a basis of $\mathbb{R}^2$: by the incomplete basis theorem (axiom of choice) or $\langle 1, 0\rangle$, $\langle 0, 1\rangle$

Why is this ZF / ZFC interoperability possible?

Because ZF and ZFC are expressed in the same logical framework: predicate logic

In predicate logic, a theory is expressed with several axioms

Permits to raise the question: which axioms are used in the proof π

The revolution of logical frameworks

Since Euclid: geometry, arithmetic, set theory... each system its language, its notion of proof...

Hilbert and Ackermann (1928): predicate logic

A logical framework in which we can define theories (geometry, arithmetic, set theory...)
For each one: symbols and axioms

A revolution that did not last long

No expression of Russell's type theory in predicate logic

1940 (Church): a new formulation of Type theory difficult (impossible?) to express in predicate logic

1970, 1985... Martin-Löf's type theory, the Calculus of constructions... not in predicate logic

Three attitudes

- Consider that the notion of logical framework is dead
- Express Russell's type theory, Church's, Martin-Löf's, the Calculus of constructions... in predicate logic by will or by force (Henkin, Davis, D...)
- Extend predicate logic in a better logical framework

Limits of predicate logic

- No bound variables ($x \mapsto x$)
- No good syntax for proofs
- No notion of computation
- No good notion of cut
- Classic and not constructive

New logical frameworks

- No bound variables ($x \mapsto x$): λ-Prolog, $\lambda\Pi$-calculus
- No good syntax for proofs: $\lambda\Pi$-calculus
- No notion of computation: Deduction modulo theory
- No good notion of cut: Deduction modulo theory
- Classic and not constructive: Ecumenical logic

The $\lambda\Pi$-calculus modulo theory that generalizes them all
An implementation: Dedukti

Simple type theory (HOL Light, Isabelle/HOL...)

$$
\begin{array}{rcl}
\mathit{type} & : & \mathit{Type} \\
\eta & : & \mathit{type} \to \mathit{Type} \\
o & : & \mathit{type} \\
\mathit{nat} & : & \mathit{type} \\
\mathit{arrow} & : & \mathit{type} \to \mathit{type} \to \mathit{type} \\
\varepsilon & : & (\eta\ o) \to \mathit{Type} \\
\Rightarrow & : & (\eta\ o) \to (\eta\ o) \to (\eta\ o) \\
\forall & : & \Pi a : \mathit{type}\ (((\eta\ a) \to (\eta\ o)) \to (\eta\ o))
\end{array}
$$

$$
\begin{array}{rcl}
(\eta\ (\mathit{arrow}\ x\ y)) & \longrightarrow & (\eta\ x) \to (\eta\ y) \\
(\varepsilon\ (\Rightarrow\ \ x\ y)) & \longrightarrow & (\varepsilon\ x) \to (\varepsilon\ y) \\
(\varepsilon\ (\forall\ x\ y)) & \longrightarrow & \Pi z : (\eta\ x)\ (\varepsilon\ (y\ z))
\end{array}
$$

The Calculus of constructions (Coq, Lean, Matita...)

$$
\begin{array}{rcl}
\mathit{type} & : & \mathit{Type} \\
\eta & : & \mathit{type} \to \mathit{Type} \\
o & : & \mathit{type} \\
\mathit{nat} & : & \mathit{type} \\
\mathit{arrow} & : & \Pi x : \mathit{type}\ (((\eta\ x) \to \mathit{type}) \to \mathit{type}) \\
\varepsilon & : & (\eta\ o) \to \mathit{Type} \\
\Rightarrow & : & \Pi x : (\eta\ o)\ (((\varepsilon\ x) \to (\eta\ o)) \to (\eta\ o)) \\
\forall & : & \Pi x : \mathit{type}\ (((\eta\ x) \to (\eta\ o)) \to (\eta\ o)) \\
\pi & : & \Pi x : (\eta\ o)\ (((\varepsilon\ x) \to \mathit{type}) \to \mathit{type})
\end{array}
$$

$$
\begin{array}{rcl}
(\eta\ (\mathit{arrow}\ x\ y)) & \longrightarrow & \Pi z : (\eta\ x)\ (\eta\ (y\ z)) \\
(\varepsilon\ (\Rightarrow\ x\ y)) & \longrightarrow & \Pi z : (\varepsilon\ x)\ (\varepsilon\ (y\ z)) \\
(\varepsilon\ (\forall\ x\ y)) & \longrightarrow & \Pi z : (\eta\ x)\ (\varepsilon\ (y\ z)) \\
(\eta\ (\pi\ x\ y)) & \longrightarrow & \Pi z : (\varepsilon\ x)\ (\eta\ (y\ z))
\end{array}
$$

Reverse mathematics in Dedukti

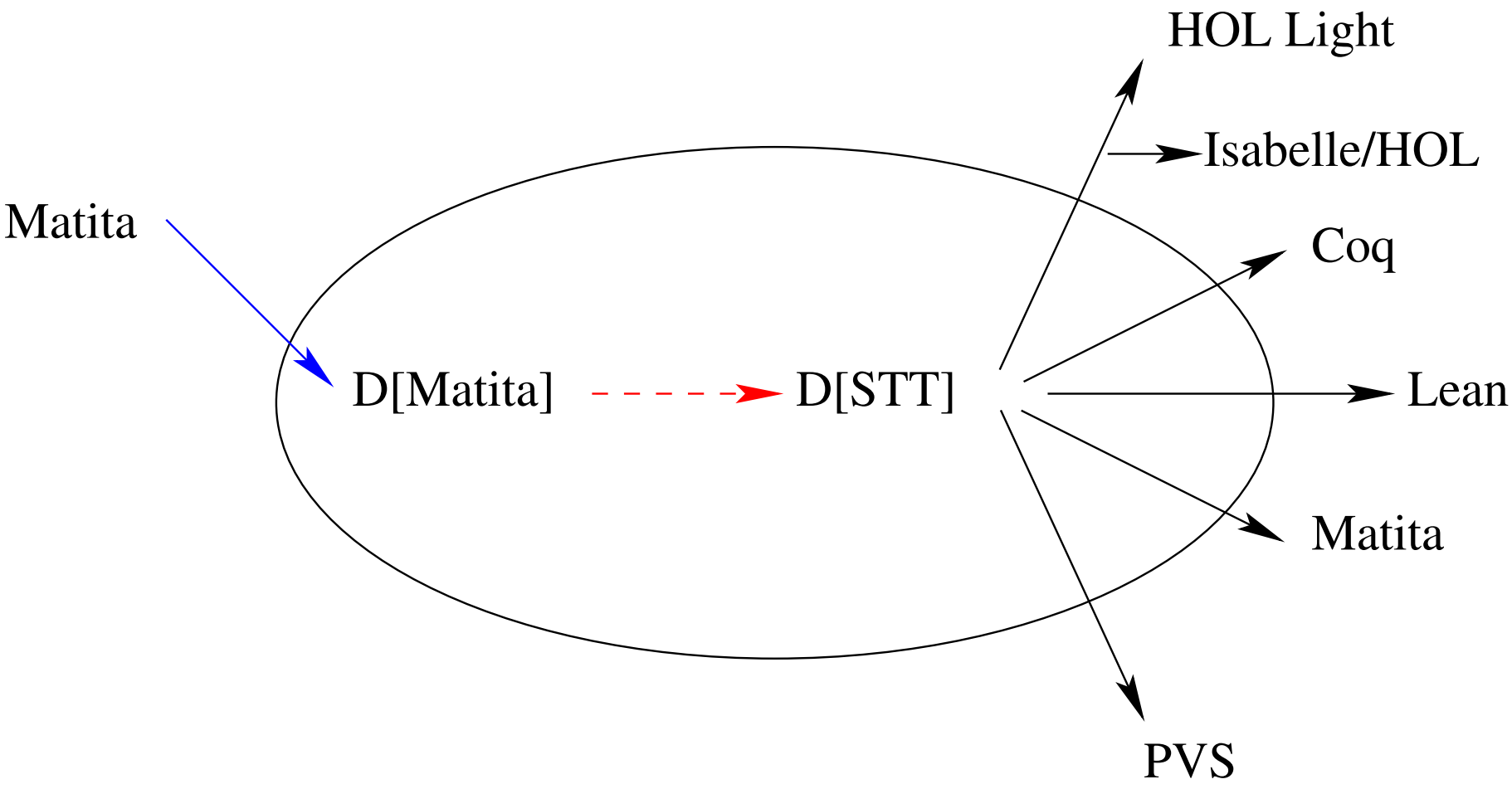
Three more features in the Calculus of constructions

All the proofs expressed in Simple type theory can be translated in the Calculus of constructions

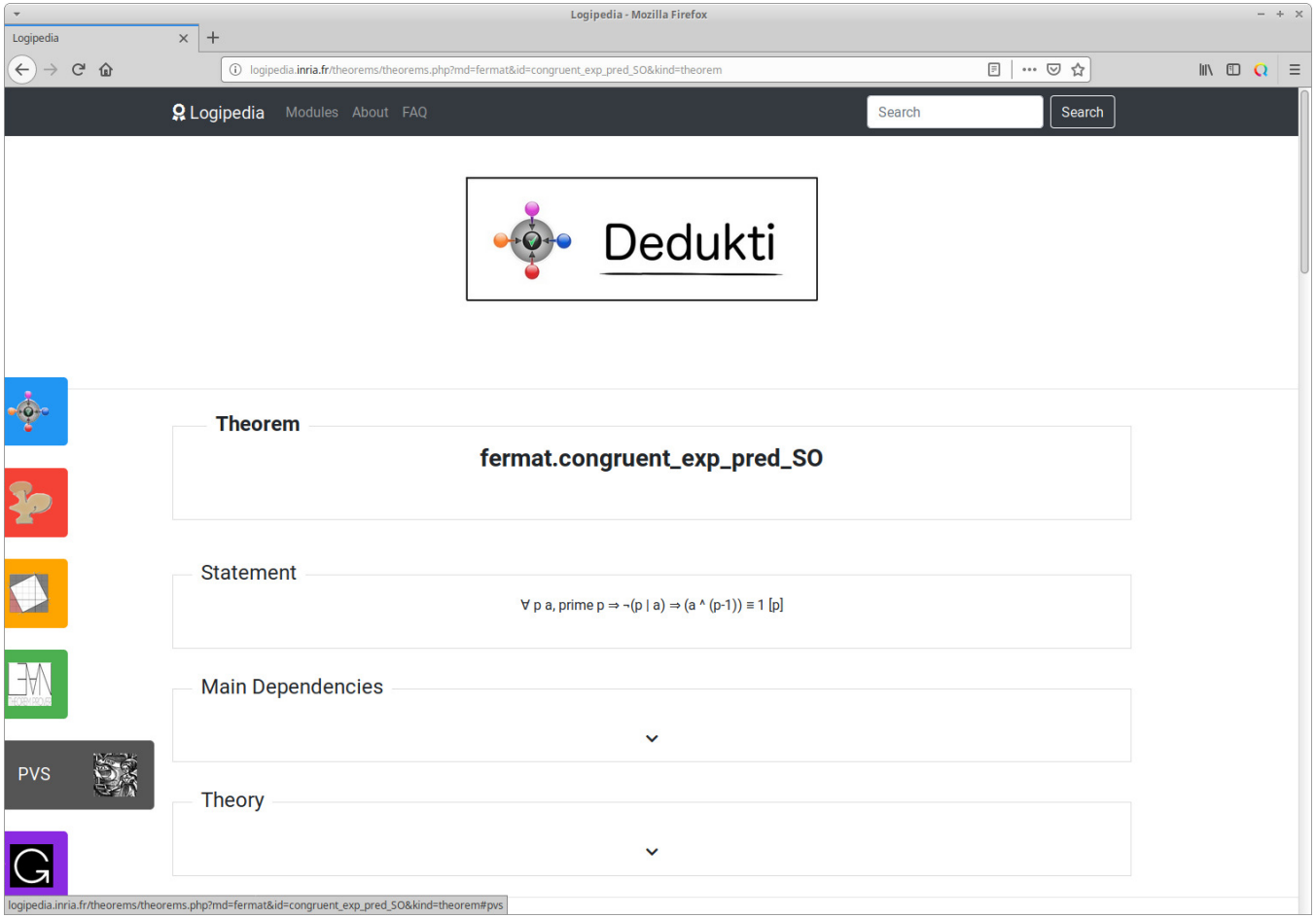
The proofs expressed in the Calculus of constructions that do not use these features can be translated in Simple type theory

For example: all the proofs of the Matita arithmetic library (Thiré)

Interoperability



Collect all these proofs in a single data base



http://logipedia.science

Why does it work?

Because proof systems implement very expressive theories of which only a small part is used in actual proofs

Previous empirical evidences

- Proof systems: very different theories, nut very similar libraries
- Mathematicians are not very interested in the axioms used in their proofs: any theory seems to fit

Interoperability is not only a question of committees, negotiations and standards. It is a research problem