The HOL Light library of formalized mathematics

Marco Maggesi – University of Florence (Italy)

EuroProofNet Workshop on the development, maintenance, refactoring and search of large libraries of proofs

Tbilisi, September 2022

Plan of this talk

1. The HOL Light theorem prover

2 The mathematical library of HOL Light

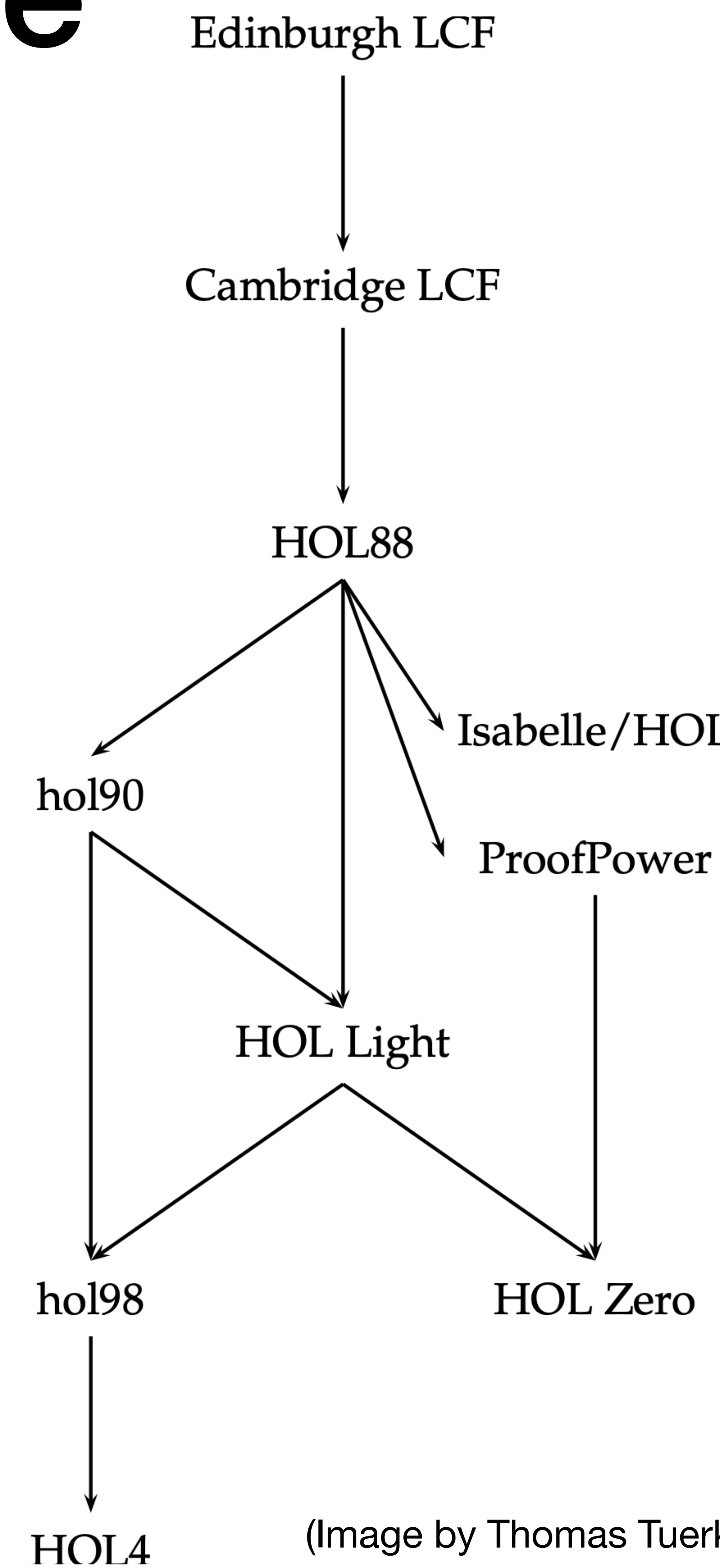
3. Relation with other systems

4. Final considerations

The HOL Light theorem prover

Historical perspective on HOL Light

- Porting of an early version of HOL from SML to CAML Light
- Prominently developed by only one author: John Harrison
- Initially focused on real analysis and, more generally, on pure mathematics



(Image by Thomas Tuerk)

The logic implemented by HOL Light

Its syntax is simple type theory:

- lambda calculus with polymorphic types

Small kernel:

- 10 primitive inference rules
- 3 axioms
- 2 principle of definitions (*nominal definition*)
- kernel of ~400 LOC

Tools:

- Inductive predicates/types, quotient types, recursive definitions, …
- Rewriting and computation
- All implemented outside of the logical kernel

Comparison with other logics/systems

- Hilbert-style logic (no Curry-Howard correspondence)
- Classical logic (Axiom of choice)
- Conceptually very easy to model in ZFC.
- Weaker than ZFC and CIC.

Very lean meta-logical or extra-logical mechanisms

- No modules/locales (no dedicated mechanisms for massive abstractions). No namespaces.
- No axiomatic classes.
- No coercions, implicit arguments.
  - Example: Euler formula $e^{i\pi} + 1 = 0$:

    ```
    cexp(ii * Cx(pi)) + Cx(&1) = Cx(&0)
    ```

  - Example: Distributive law in ring theory

    ```
    |- !r x y z:A.
         x IN ring_carrier r /\
         y IN ring_carrier r /\
         z IN ring_carrier r
         ==> ring_mul r x (ring_add r y z) =
             ring_add r (ring_mul r x y) (ring_mul r x z)
    ```

The mathematical library of HOL Light

Content of the library

Impressive library (especially for pure mathematics), encompassing most of the central topics in classical mathematics:

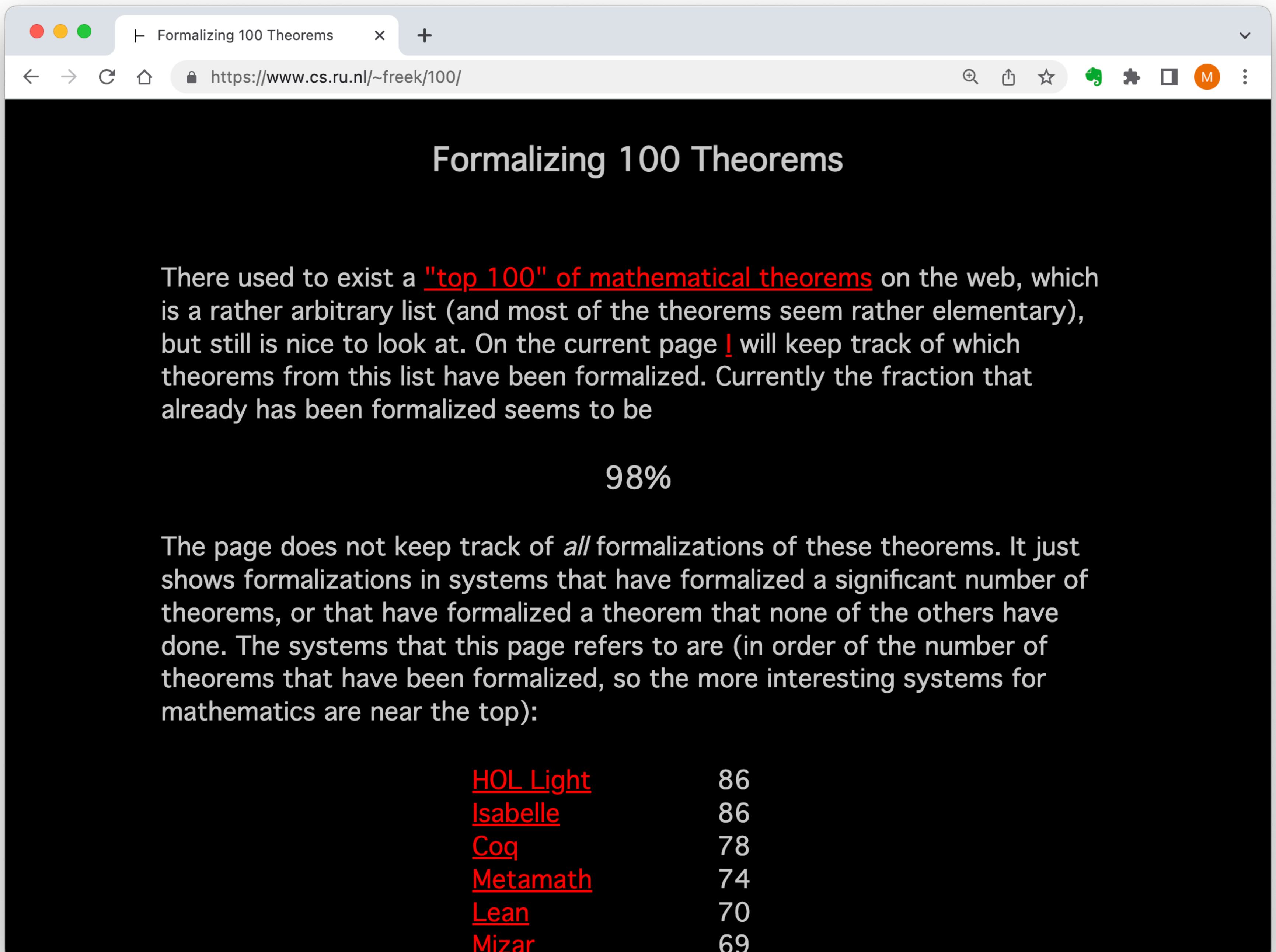
- Real multivariate analysis, including linear algebra, topology and metric spaces, integration, algebraic topology and degree theory and more, …
- Complex analysis (including some quaternionic/hypercomplex analysis), up to advanced theorems such us the Riemann uniformisation theorem and the applications to analytic number theory (asymptotic distribution of primes).
- Algebra, e.g. group theory and ring theory.
- Elliptic curves (cryptography).
- Logic, e.g., Gödel incompleteness theorems, Gödel-Löb provability theory.
- …
- Hales' Theorem (Kepler conjecture).

Topics that are missing in the HOL Light library

The HOL Light library misses (to my best knowledge) certain classical topics that are available in many other systems:

- Category theory.
- Universal Algebra.
- Data structures.
- Probability theory (but has a lot about measure theory).
- Computability theory, complexity theory.
- Lambda calculus (but there is a deep embeddings of HOL and HOL and work on initial semantics with De Bruijn encoding).
- Theories and tools for advanced (co)inductive types and (co)recursion.
- Set theory.
- …

“Top 100” of mathematical theorems



Formalizing 100 Theorems

https://www.cs.ru.nl/~freek/100/

Formalizing 100 Theorems

There used to exist a "top 100" of mathematical theorems on the web, which is a rather arbitrary list (and most of the theorems seem rather elementary), but still is nice to look at. On the current page I will keep track of which theorems from this list have been formalized. Currently the fraction that already has been formalized seems to be

98%

The page does not keep track of *all* formalizations of these theorems. It just shows formalizations in systems that have formalized a significant number of theorems, or that have formalized a theorem that none of the others have done. The systems that this page refers to are (in order of the number of theorems that have been formalized, so the more interesting systems for mathematics are near the top):

HOL Light	86
Isabelle	86
Coq	78
Metamath	74
Lean	70
Mizar	69

Some random observations

- Consistent coding style.
- Very high-quality code
(both for the humans and for the machines).
- Very conservative development, strict backward compatibility with few exceptions over many years.
- Many good tactics and decision procedure available.

Relation with other systems

Work derived from HOL Light

- **HOL Zero** by M. Adams

  (Pollack-consistent variant of HOL Light)

- **HOL Light QE** by J. Carette, W. M. Farmer, P. Laskowski

  (Quotation and Evaluation)

- **HOL Light in λProlog** by C. Dunchev, C. Sacerdoti Coen, E. Tassi

  (HOL “super light” in ELPI)

- **HOL2P** by N. Völker

  (Second Order Polymorphism)

Exporting from HOL Light

HOL Light has a few of proof recording mechanisms:

Proof Recording

Attach proof objects to HOL theorems.

Used for the exportation mechanism to Coq made by S. Obua and C. Keller.

Proof Trace

Records the proof steps of the HOL kernel.

Developed by S. Polu, used for Machine Learning experiments.

OpenTheory

Export theories in a standardised format.

Developed by J. Hurd, can export/import theorems from other HOL systems such as HOL4 and ProofPower

...

HOL Light and Machine Learning

- **HOL(y) Hammer** by C. Kaliszyk and J. Urban

  (ATP with machine-learning premise selection)

- **HOList / DeepHOL** by K. Bansal, et al.

  (Reinforcement learning)

Final considerations

I use HOL Light when I want to …

- … experiment with certain **advanced mathematical topics** such as (hyper)complex analysis. E.g.
  - a formalisation of **Cartan rigidity theorems** (in collaboration with G. Ciolli and G. Gentili)
  - a formalisation of **slice regular quaternionic functions** (in collaboration with A. Gabrielli)
- … show the **conceptual simplicity** of a formalisation approach. E.g.:
  - a recent work on **De Bruijn encoding** (with A, Hirschowitz, T. Hirshowitz, A. Lafont)
- … develop a **tactic/decision procedure**. E.g:
  - a decision procedure for **metric spaces**
  - a decision procedure for the **Gödel-Löb logic** of probability GL (in collaboration with C. Perini Brogi)

Final considerations

- HOL Light features a *“simple”* logic foundation and comes with an interesting library of mathematical results.
- Elaborations on this collection of mathematical results might be not too difficult conceptually and may eventually lead to several interesting applications.
- However, the “knowledge” present in the system goes far beyond the mere collection of theorems.
- For certain more ambitions goals, we must consider HOL Light endowed with a much richer meta-logic (eventually including the full programming language ML).