

Synthetic Mathematics for the Mechanisation of Computability Theory and Logic

Yannick Forster  

Inria Paris, France

Abstract

Mathematical practice in most areas of mathematics is based on the assumption that proofs could be made fully formal in a chosen foundation in principle. This assumption is backed by partial attempts at formalisation and by full mechanisation of various areas of mathematics in various proof assistants and various foundations. Areas that have been largely neglected for computer-assisted and machine-checked proofs are computability theory and logic: Fundamental results like Gödel’s second incompleteness theorem in its stronger forms due to Kleene and Rosser, Löb’s theorem, Post’s theorem connecting the arithmetical hierarchy and Turing jumps, or the Friedberg-Mučnik theorem solving Post’s problem have not or only very recently been re-produced in proof assistants. This is due to the fact that making these arguments formal is several orders of magnitude more involved than formalising other areas of mathematics, due to the amount of invisible mathematics (a term coined by Andrej Bauer) involved.

In computability theory, invisible arguments occur mainly behind proofs that a certain intuitively sketched procedure is computable in – citing Emil Post – “forbidding, diverse and alien formalisms in which this [...] work of Gödel, Church, Turing, Kleene, Rosser [...] is embodied.” For instance, there have been various approaches of formalising Turing machines, all to the ultimate dissatisfaction of the respective authors, and none going further than constructing a universal machine and proving the halting problem undecidable. Professional computability theorist and teachers of computability theory thus rely on the informal Church Turing thesis to carry out their work and only argue the computability of described algorithms informally.

For computability theory, a way out was proposed in the 1980s by Fred Richman and developed during the last decade by Andrej Bauer: Synthetic computability theory, where one assumes axioms in a constructive foundation which essentially identify all (constructively definable) functions with computable functions. A drawback of the approach is that assuming such an axiom on top of the axiom of countable choice - which is routinely assumed in this branch of constructive mathematics and computable analysis - is that the law of excluded middle, i.e. classical logic, becomes invalid. Computability theory is however, as all mainstream branches of mathematics, making routine use of the axiom of excluded middle.

In the case of logic, the invisible mathematics usually is either centered around encoding formulas and proofs as numbers using Gödel or similar encodings or about provability arguments that certain results can be proved in restricted proof systems such as Peano arithmetic. In several settings, synthetic computability arguments can be employed to mechanise these proofs.

We observe that a slight foundational shift rectifies the situation: By basing synthetic computability theory in the Calculus of Inductive Constructions, the type theory underlying amongst others the Coq and Lean proof assistants, where countable choice is independent and thus not provable, axioms for synthetic computability are compatible with the law of excluded middle. This insight can be used to finally mechanise computability theory and logic, in an elegant, concise way where invisible arguments stay invisible: with Felix Jahn I have mechanised arguments related to many-one and truth-table reduction theory (published at CSL ’23), Dominik Kirst and Benjamin Peters have presented Gödel’s first incompleteness theorem in this style (at CSL ’23), and in collaboration with Dominik Kirst and Niklas Mück I have given a proof of Post’s hierarchy theorem (at CSL ’24).

In this invited talk, I will give a broader overview of this line of research investigating a mechanised synthetic approach to logic and computability theory. In particular, I will discuss a Coq library of undecidability proofs, results in the theory of reducibility degrees, constructive reverse analysis of theorems, as well as generalised incompleteness results such as Löb’s theorem.



© Yannick Forster;

licensed under Creative Commons License CC-BY 4.0

33rd EACSL Annual Conference on Computer Science Logic (CSL 2025).

Editors: Jörg Endrullis and Sylvain Schmitz; Article No. 3; pp. 3:1–3:2

Leibniz International Proceedings in Informatics



LIPIC Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

3:2 Synthetic Mathematics for the Mechanisation of Computability Theory and Logic

2012 ACM Subject Classification Theory of computation → Constructive mathematics

Keywords and phrases Synthetic mathematics, computability theory, logic

Digital Object Identifier 10.4230/LIPIcs.CSL.2025.3

Category Invited Talk

Supplementary Material

Software: <https://github.com/uds-psl/coq-synthetic-computability/>

Software: <https://github.com/uds-psl/coq-library-fo1>

Acknowledgements Joint work with Dominik Kirst, Gert Smolka, Felix Jahn, Niklas Mück, Janis Bailitis, Haoyi Zeng, and the contributors of the Coq Library of Undecidability Proofs.