Löb's Theorem and Provability Predicates in Rocq

Janis Bailitis[1,3], Dominik Kirst[2], and Yannick Forster[2]

[1] Saarland University, Saarbrücken, Germany
[2] Inria, Paris, France
[3] University of Oxford, Oxford, U.K.

Löb's theorem ([24], short: LT) states that in sufficiently strong formal systems such as Peano Arithmetic (PA), for a sentence φ we have $\mathsf{PA} \vdash \varphi$ if and only if $\mathsf{PA} \vdash \mathsf{prov}_{\mathsf{PA}}[\ulcorner\varphi\urcorner] \dot{\rightarrow} \varphi$, where the formula $\mathsf{prov}_{\mathsf{PA}}(x)$ is the standard provability predicate. It is a strengthening of Gödel's second incompleteness theorem (short: G2) which can be recovered via $\varphi := \bot$. Similar to the incompleteness theorems, the proof of LT is highly technical. Even for a fixed system such as PA, there are many different provability predicates of varying strengths. Not all of them qualify for LT, and formal reasoning about provability predicates is highly tedious.

For Gödel's first incompleteness theorem (short: G1), even in Rosser's [31] strengthening, these technical challenges can be avoided, as demonstrated by Kirst and Peters [20] who mechanise an abstract and computational proof of G1 [12] due to Kleene [21, 22] in Rocq [34]. They build their proof on the axiom Church's Thesis (CT) [23, 35], a well-understood axiom in constructive mathematics stating that quantifiers over functions in a constructive setting only range over computable functions. Fundamentally, their proof reduces G1 to the undecidability of provability in PA, mechanised by Kirst and Hermes [18].

For this abstract, we

1. mechanise the traditional proof of G1 via Carnap's diagonal lemma [6], using CT, with Rosser's [31] strengthening, complementing Kirst and Peters' computational mechanisation,
2. mechanise Tarski's theorem [33] about the undefinability of truth,
3. mechanise a proof of LT parameterised against a sufficiently strong provability predicate,
4. define a provability predicate and prove some of the necessary properties, all in Rocq,
5. extend Paulson's [28, 27] Isabelle mechanisation of a sufficiently strong predicate to yield LT.

In general, the abstract approach of Kirst and Peters does not extend to G2 or LT, because these theorems inherently rely on concrete implementation details of the underlying logical system, unlike G1. The results of this paper mechanised in the Rocq Prover [34] formally work in the Calculus of Inductive Constructions (CIC) [7, 26]. They rely on and are contributed to the Rocq Library for First-Order Logic [19] and the Rocq Library of Undecidability Proofs [11]. All proofs are constructive. This abstract is based on the first author's Bachelor's thesis [1] carried out in the group of Gert Smolka, advised by the other authors.

Mechanised synthetic computability. We use *synthetic computability theory* due to Richman, Bridges, and Bauer [29, 5, 2]. In synthetic computability theory, the usual notions from computability theory are defined without referring to a concrete model of computation. For instance, a predicate $P : X \to \mathbb{P}$ is said to be decidable iff there is a decider $f : X \to \mathbb{B}$ such that for all x, $P\,x$ holds iff $f\,x = \mathsf{true}$. If $\mathcal{O}(X) ::= \mathsf{Some}(x) \mid \mathsf{None}$ denotes the option type over X, we say that P is (recursively) enumerable if there is an enumerator $f : \mathbb{N} \to \mathcal{O}(X)$ such that for all x, $P\,x$ holds iff there exists n such that $f\,n = \mathsf{Some}(x)$. In type theory, synthetic computability has been developed by Forster [9] and colleagues. Of importance for this abstract is the standard fact [10] that $\lambda\varphi.\, T \vdash \varphi$ is (recursively) enumerable if T is.

Arithmetical Church's thesis. We define a variant of Church's thesis for Robinson arithmetic (Q) [30], a subsystem of PA. A formula φ is Σ_1 if it is of the form $\dot\exists x_1. \ldots. \dot\exists x_n. \psi$ and ψ does not use unbounded quantification. A theory T is called Σ_1-sound if $T \vdash \varphi$ implies $\mathbb{N} \vDash \varphi$ for all Σ_1-formulas φ. Church's thesis for Robinson arithmetic ($\mathsf{CT_Q}$) states that for every function $f : \mathbb{N} \to \mathbb{N}$, there exists a binary Σ_1-formula $\varphi(x, y)$ such that for all n, it holds that $\mathsf{Q} \vdash \dot\forall y. \varphi[\overline{n}, y] \dot\leftrightarrow y \dot= \overline{f\,n}$. $\mathsf{CT_Q}$ is already employed by Kirst, Hermes, and Peters [14, 15, 20]. If applied to deciders and enumerators, $\mathsf{CT_Q}$ implies that certain predicates can be represented in Q as well [20, 14]. Most importantly, for any (recursively) enumerable $P : X \to \mathbb{P}$ and any Σ_1-sound $T \supseteq \mathsf{Q}$, there is a unary Σ_1-formula $\varphi(x)$ such that for all n, $P\,n$ holds iff $T \vdash \varphi[\overline{n}]$ (*weak representation*). Also, for any disjoint (recursively) enumerable $P, P' : X \to \mathbb{P}$ and consistent $T \supseteq \mathsf{Q}$, there is a unary Σ_1-formula $\varphi(x)$ such that for all n, both $P\,n$ implies $T \vdash \varphi[\overline{n}]$ and $P'\,n$ implies $T \vdash \dot\neg\varphi[\overline{n}]$ (*strong separation*).

Diagonal lemma and G1. A standard approach [4, 32] to prove G1 is to establish the diagonal lemma [6], stating that for all unary formulas $\varphi(x)$ there is a sentence G such that $\mathsf{Q} \vdash \varphi[\ulcorner G \urcorner] \dot\leftrightarrow G$, where $\ulcorner\cdot\urcorner$ is an encoding of formulas into closed terms. As a second step, the diagonal lemma is used to diagonalise against a provability predicate. We observe that the diagonal lemma readily follows from $\mathsf{CT_Q}$. G1 then easily follows from (recursive) enumerability of $\lambda\varphi. T \vdash \varphi$ with a brief application of weak representability and diagonalisation. We also mechanise the following strengthening of G1 following Rosser [31].

Theorem 1 (G1 [12])**.** *Let $T \supseteq \mathsf{Q}$ be (recursively) enumerable and consistent. Then there is an independent sentence for T.*

Here, the idea is to diagonalise against the negation of the formula obtained from strong separability applied to $\lambda\varphi. T \vdash \varphi$ and $\lambda\varphi. T \vdash \dot\neg\varphi$. Similar reasoning also gives rise to Tarski's theorem [33].

Theorem 2 (Tarski [33])**.** *There is no first-order formula $\mathsf{true}_{\mathbb{N}}(x)$ such that $\mathbb{N} \vDash \varphi$ iff $\mathbb{N} \vDash \mathsf{true}_{\mathbb{N}}[\ulcorner\varphi\urcorner]$ for all sentences φ.*

For all these results, $\mathsf{CT_Q}$ is extremely helpful since instead of defining actual formulas for the provability predicates, only enumerability of provability is needed, which is easy to establish synthetically. The proofs thus reduce to the key insights gained from Gödel's and Tarski's work, making the proofs extremely concise. Without $\mathsf{CT_Q}$, the results can still be shown if one assumes that T is μ-recursively enumerable, but then the proofs would become very tedious.

LT and internal vs external provability. Following a classification due to Feferman [8], we distinguish between *external* and *internal* provability predicates. An external provability predicate only has to correctly identify provable formulas, i.e. $T \vdash \varphi$ iff $T \vdash \mathsf{prov}_T[\ulcorner\varphi\urcorner]$ for all φ. $\mathsf{CT_Q}$ implies the existence of an external provability predicate, which is sufficient for Theorem 1. Mostowski [25], Bezboruah, and Shepherdson [3] observe that external predicates are too weak for LT by giving an external predicate for which G2 and hence also LT fails.

An internal provability predicate needs to additionally allow proving some deduction rules of the logical system as object level implications. This was made precise by Löb [24], based on previous work by Hilbert and Bernays [16]. The required properties are known as Hilbert-Bernays-Löb (HBL) derivability conditions, which are as follows, where φ, ψ are arbitrary formulas:

$$T \vdash \varphi \text{ implies } T \vdash \mathsf{prov}_T[\ulcorner\varphi\urcorner] \qquad \text{(necessitation)}$$

$$T \vdash \mathsf{prov}_T[\ulcorner\varphi \dot\to \psi\urcorner] \dot\to \mathsf{prov}_T[\ulcorner\varphi\urcorner] \dot\to \mathsf{prov}_T[\ulcorner\psi\urcorner] \qquad \text{(modus ponens rule)}$$

$$T \vdash \mathsf{prov}_T[\ulcorner\varphi\urcorner] \dot\to \mathsf{prov}_T[\mathsf{prov}_T[\ulcorner\varphi\urcorner]] \qquad \text{(internal necessitation)}$$

For an internal provability predicate, LT follows abstractly, and we mechanise this abstract proof in Rocq.

Theorem 3 (Löb [24]). *Let T be a theory admitting the diagonal lemma and let $\mathsf{prov}_T(x)$ satisfy the HBL conditions. Then $T \vdash \varphi$ iff $T \vdash \mathsf{prov}_T[\ulcorner \varphi \urcorner] \to \varphi$ for all sentences φ.*

Defining internal provability predicates. To do so, most authors arithmetise the notion of provability and define a complicated formula $\mathsf{prf}_T(x, y)$ such that $\mathsf{prf}_T(x, y)$ is provable iff y arithmetises a proof of the formula with code x. In this setting $\dot{\exists} y.\, \mathsf{prf}_T(x, y)$ characterises provability of x. Usually, a proof of a formula φ is encoded as a list of formulas representing the deduction of φ from the deduction rules. This is also how the standard provability predicate for PA is constructed.

While PA and related systems of arithmetic are strong enough to express the required list functions and to prove properties about these functions on the object level, choosing arithmetic to define provability predicates is not ideal. Instead, we prove the following:

Theorem 4. *There is an extension of* PA *with native function symbols for basic list functions to avoid some of the technicalities. In this system, there is an internal provability predicate satisfying necessitation and the modus ponens rule.*

This development is axiom-free. The verification of necessitation and the modus ponens rule heavily relies on a proof mode for first-order logic due to Koch [17], which made these mechanisations possible in the first place. As part of this proof, we contribute a mechanisation of Hilbert systems and a proof of its equivalence to natural deduction to the Rocq Library of First-Order Logic [19].

Paulson [28, 27] mechanises an internal provability predicate in HF set theory, easing arithmetisation, but the definition and the correctness proofs are still very arduous. We add the following to Paulson's Isabelle development:

Theorem 5. *Paulson's provability predicate is sufficient to deduce LT.*

Future work. Ultimately, we would like to obtain a Rocq mechanisation and verification of an internal provability predicate in a suitable system of first-order logic resembling PA. Besides doing the tedious work to prove internal necessitation for our predicate, it also seems promising to follow the approach by Halbach and Leigh [13] who give a system of first-order logic with function symbols for syntax manipulation which can express PA. These syntax functions seem helpful in the verification of internal necessitation.

In addition, it may be desirable to understand how strong a theory T of first-order arithmetic needs to be such that the HBL conditions for T's standard provability predicate can be proved. There may be research on this of which the authors are currently unaware.

References

[1] Janis Bailitis. Löb's Theorem and Provability Predicates in Coq, 2024. Bachelor's thesis. URL: https://www.ps.uni-saarland.de/~bailitis/bachelor.php.

[2] Andrej Bauer. First Steps in Synthetic Computability Theory. In Martín Hötzel Escardó, Achim Jung, and Michael W. Mislove, editors, *Proceedings of the 21st Annual Conference on Mathematical Foundations of Programming Semantics, MFPS 2005, Birmingham, UK, May 18-21, 2005*, volume 155 of *Electronic Notes in Theoretical Computer Science*, pages 5–31. Elsevier, 2005. doi:10.1016/J.ENTCS.2005.11.049.

[3] A. Bezboruah and John C. Shepherdson. Gödel's Second Incompleteness Theorem for Q. *The Journal of Symbolic Logic*, 41(2):503–512, 1976. doi:10.1017/S0022481200051586.

[4] George S. Boolos, John P. Burgess, and Richard C. Jeffrey. *Computability and Logic*. Cambridge University Press, 5th edition, 2007.

[5] Douglas Bridges and Fred Richman. *Varieties of Constructive Mathematics*. London Mathematical Society Lecture Note Series. Cambridge University Press, 1987.

[6] Rudolf Carnap. *Logische Syntax der Sprache*. Schriften zur wissenschaftlichen Weltauffassung. Springer Berlin, Heidelberg, 1st edition, 1934.

[7] Thierry Coquand and Gérard P. Huet. The Calculus of Constructions. *Information and Computation*, 76(2/3):95–120, 1988. doi:10.1016/0890-5401(88)90005-3.

[8] Solomon Feferman. Arithmetization of metamathematics in a general setting. *Fundamenta mathematicae*, 49:35–92, 1960.

[9] Yannick Forster. *Computability in Constructive Type Theory*. PhD thesis, Saarland University, 2021. doi:10.22028/D291-35758.

[10] Yannick Forster, Dominik Kirst, and Gert Smolka. On Synthetic Undecidability in Coq, with an Application to the Entscheidungsproblem. In Assia Mahboubi and Magnus O. Myreen, editors, *Proceedings of the 8th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2019, Cascais, Portugal, January 14-15, 2019*, pages 38–51. ACM, 2019. doi:10.1145/3293880.3294091.

[11] Yannick Forster, Dominique Larchey-Wendling, Andrej Dudenhefner, Edith Heiter, Dominik Kirst, Fabian Kunze, Gert Smolka, Simon Spies, Dominik Wehr, and Maximilian Wuttke. A Coq library of undecidable problems. *CoqPL 2020: The Sixth International Workshop on Coq for Programming Languages*, 2020.

[12] Kurt Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik*, 38(1):173–198, 1931.

[13] Volker Halbach and Graham E. Leigh. *The Road to Paradox*. Cambridge University Press, 2024. doi:10.1017/9781108888400.

[14] Marc Hermes and Dominik Kirst. An Analysis of Tennenbaum's Theorem in Constructive Type Theory. In Amy P. Felty, editor, *7th International Conference on Formal Structures for Computation and Deduction (FSCD 2022)*, volume 228 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 9:1–9:19, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.FSCD.2022.9.

[15] Marc Hermes and Dominik Kirst. An Analysis of Tennenbaum's Theorem in Constructive Type Theory. *CoRR*, abs/2302.14699, 2023. arXiv:2302.14699, doi:10.48550/ARXIV.2302.14699.

[16] David Hilbert and Paul Bernays. *Grundlagen der Mathematik*, volume 2. Springer, Berlin, 1st edition, 1939.

[17] Johannes Hostert, Mark Koch, and Dominik Kirst. A Toolbox for Mechanised First-Order Logic. *The Coq Workshop*, 2021.

[18] Dominik Kirst and Marc Hermes. Synthetic Undecidability and Incompleteness of First-Order Axiom Systems in Coq. *Journal of Automated Reasoning*, 67(1):13, 2023. doi:10.1007/S10817-022-09647-X.

[19] Dominik Kirst, Johannes Hostert, Andrej Dudenhefner, Yannick Forster, Marc Hermes, Mark Koch, Dominique Larchey-Wendling, Niklas Mück, Benjamin Peters, Gert Smolka, and Dominik Wehr. A Coq Library for Mechanised First-Order Logic. *The Coq Workshop*, 2022.

[20] Dominik Kirst and Benjamin Peters. Gödel's Theorem Without Tears - Essential Incompleteness in Synthetic Computability. In Bartek Klin and Elaine Pimentel, editors, *31st EACSL Annual Conference on Computer Science Logic (CSL 2023)*, volume 252 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 30:1–30:18, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CSL.2023.30.

[21] Stephen C. Kleene. Recursive Predicates and Quantifiers. *Transactions of the American Mathematical Society*, 53:41–73, 1943. doi:10.2307/2267986.

[22] Stephen C. Kleene. *Mathematical Logic*. Dover Publications, 1967.

[23] Georg Kreisel. Mathematical logic. *Lectures on Modern Mathematics*, 3:95–195, 1965.

[24] Martin H. Löb. Solution of a Problem of Leon Henkin. *The Journal of Symbolic Logic*, 20(2):115–118, 1955. doi:10.2307/2266895.

[25] Andrzej Mostowski. Thirty years of foundational studies: lectures on the development of mathematical logic and the study of the foundations of mathematics in 1930-1964. *Acta Philosophica Fennica*, 17:1–180, 1965.

[26] Christine Paulin-Mohring. Inductive Definitions in the system Coq - Rules and Properties. In Marc Bezem and Jan Friso Groote, editors, *Typed Lambda Calculi and Applications, International Conference on Typed Lambda Calculi and Applications, TLCA '93, Utrecht, The Netherlands, March 16-18, 1993, Proceedings*, volume 664 of *Lecture Notes in Computer Science*, pages 328–345. Springer, 1993. doi:10.1007/BFB0037116.

[27] Lawrence C. Paulson. A Machine-Assisted Proof of Gödel's Incompleteness theorems for the Theory of Hereditarily Finite Sets. *The Review of Symbolic Logic*, 7(3):484–498, 2014. doi:10.1017/S1755020314000112.

[28] Lawrence C. Paulson. A Mechanised Proof of Gödel's Incompleteness Theorems Using Nominal Isabelle. *Journal of Automated Reasoning*, 55(1):1–37, 2015. doi:10.1007/S10817-015-9322-8.

[29] Fred Richman. Church's Thesis Without Tears. *The Journal of Symbolic Logic*, 48(3):797–803, 1983. doi:10.2307/2273473.

[30] Raphael Robinson. An essentially undecidable axiom system. *Proceedings of the International Congress of Mathematics*, pages 729–730, 1950.

[31] J. Barkley Rosser. Extensions of Some Theorems of Gödel and Church. *The Journal of Symbolic Logic*, 1(3):87–91, 1936. doi:10.2307/2269028.

[32] Peter Smith. *An Introduction to Gödel's Theorems*. Logic Matters, Cambridge, 2nd edition, 2020. URL: https://www.logicmatters.net/resources/pdfs/godelbook/GodelBookLM.pdf.

[33] Alfred Tarski. Der Wahrheitsbegriff in den formalisierten Sprachen. *Studia Philosophica. Commentarii Societatis Philosophicae Polonorum*, 1:261 – 405, 1935. URL: https://www.sbc.org.pl/dlibra/publication/24411/edition/21615.

[34] The Coq Development Team. The Coq Proof Assistant, September 2023. doi:10.5281/zenodo.11551177.

[35] Dirk van Dalen and Anne S. Troelstra. *Constructivism in Mathematics*. Elsevier Science Publishers B.V., 1988.