Gödel's Theorem Without Tears

Essential Incompleteness in Synthetic Computability

Dominik Kirst and Benjamin Peters

CSL'23, February 16th

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Gödel's Theorem Without Tears?

THE JOURNAL OF SYMBOLIC LOGIC
Volume 48, Number 3, Sept. 1983

CHURCH'S THESIS WITHOUT TEARS

FRED RICHMAN

§1. **Introduction.** The modern theory of computability is based on the works of Church, Markov and Turing who, starting from quite different models of computation, arrived at the same class of computable functions. The purpose of this paper is the show how the main results of the Church-Markov-Turing theory of computable functions may quickly be derived and understood without recourse to the largely irrelevant theories of recursive functions, Markov algorithms, or Turing machines. We do this by ignoring the problem of what constitutes a computable function and concentrating on the central feature of the Church-Markov-Turing theory: that the set of computable partial functions can be effectively enumerated. In this manner we are led directly to the heart of the theory of computability without having to fuss about what a computable function is.

Get to the heart of computational incompleteness proofs without having to fuss about what a computable function is!

The First Incompleteness Theorem

Which formal systems $\mathcal{S}$ admit sentences φ with both $\mathcal{S} \nvdash \varphi$ and $\mathcal{S} \nvdash \neg\varphi$?

- Gödel: all sound, sufficiently expressive ones (Gödel, 1931)
- Rosser: all consistent, sufficiently expressive ones (Rosser, 1936)
- Church/Turing(/Post): Gödel's incompleteness follows from undecidability
- Kleene: Rosser's incompleteness follows from recursive inseparability (Kleene, 1951)
- We give synthetic computational proofs complementing mechanisations à la Gödel/Rosser: Shankar (1986); O'Connor (2005); Paulson (2015); Popescu and Traytel (2019)

Synthetic Incompleteness (Kirst and Hermes, 2021)

$P : X \to \mathbb{P}$ is decidable if there exists $d : X \to \mathbb{B}$ with $P\,x \leftrightarrow d\,x = \mathrm{tt}$

$P : X \to \mathbb{P}$ is semi-decidable if there exists $s : X \to \mathbb{N} \to \mathbb{B}$ with $P\,x \leftrightarrow \exists n.\, s\,x\,n = \mathrm{tt}$

Theorem

If Robinson's Q (or any sound extension) is complete, then the halting problem is decidable.

Sketch.

Systems like Q are semi-decidable, complete ones also co-semi-decidable and hence decidable. Thus all predicates soundly captured in such a complete system are decidable. □

Shortcomings:

1. Not an explicit negation, only a computational taboo
2. No explicit independent sentence is constructed
3. Requires soundness to extract correct information from formal derivations

Stronger Synthetic Incompleteness Results

Abstract Formal Systems

Definition

A triple $\mathcal{S} = (\mathbb{S}, \neg, \vdash)$ is called a formal system if:

- $\mathbb{S}$ is a type, considered the sentences of $\mathcal{S}$
- $\neg : \mathbb{S} \to \mathbb{S}$ is a function on sentences, considered the negation operation
- $\vdash : \mathbb{S} \to \mathbb{P}$ is a semi-decidable predicate on sentences, considered the provable sentences
- Consistency holds in the form that for all $\varphi : \mathbb{S}$ not both $\vdash \varphi$ and $\vdash \neg\varphi$

Instances:

- First-order axiomatisations like Q, PA, HA, ZF, IZF, ...
- Second-order arithemtics and set theories
- Simple and dependent type theories

The Weak Church-Turing Proof

Lemma

Given a formal system $\mathcal{S} = (\mathbb{S}, \neg, \vdash)$, one can construct a partial function $d_{\mathcal{S}} : \mathbb{S} \rightharpoonup \mathbb{B}$ with:

$$d_{\mathcal{S}}\,\varphi \downarrow \mathsf{tt} \;\leftrightarrow \vdash \varphi \qquad \text{and} \qquad d_{\mathcal{S}}\,\varphi \downarrow \mathsf{ff} \;\leftrightarrow \vdash \neg\varphi$$

By this specification, $d_{\mathcal{S}}$ exactly diverges on the independent sentences of $\mathcal{S}$.

Theorem

Let $\mathcal{S}$ weakly represent $P : \mathbb{N} \to \mathbb{P}$, i.e. assume there is a function $\varphi_P : \mathbb{N} \to \mathbb{S}$ with:

$$P\,x \;\leftrightarrow\; \vdash \varphi_P(x)$$

If $\mathcal{S}$ is complete, i.e. satisfies $\vdash \varphi$ or $\vdash \neg\varphi$ for all φ, then P is decidable.

Proof.

If $\mathcal{S}$ is complete, then $d_{\mathcal{S}}$ is a total function $\mathbb{S} \to \mathbb{B}$ and $d_{\mathcal{S}} \circ \varphi_P$ is a decider for P. □

Church's Thesis

Consistent assumption in many variants of constructive mathematics:

- Kreisel (1970): "Every function can be captured by Kleene's T-predicate"
- Richman (1983): "The set of partial functions is countable"
- Bauer (2006): "There are enumerably many enumerable sets"
- Swan and Uemura (2019): consistency proof for (homotopy) type theory

Axiom (EPF, cf. Forster (2021))

There is a universal function $\Theta : \mathbb{N} \to (\mathbb{N} \rightharpoonup \mathbb{N})$ *enumerating all partial functions:*

$$\forall f : \mathbb{N} \rightharpoonup \mathbb{N}.\, \exists c : \mathbb{N}.\, \forall xy.\, \Theta_c\, x \downarrow y \leftrightarrow f\, x \downarrow y$$

Synthetic Halting Problem

Lemma

$\mathsf{K}_\Theta\, x := \Theta_x\, x \downarrow$ *is undecidable, in fact for every candidate decider* $d : \mathbb{N} \rightharpoonup \mathbb{B}$ *with*

$$\mathsf{K}_\Theta\, x \;\leftrightarrow\; d\, x \downarrow \mathsf{tt}$$

one can construct a concrete value c *such that* $d\, c$ *diverges.*

Proof.

We first define a partial function $f : \mathbb{N} \rightharpoonup \mathbb{B}$ diagonalising against d by:

$$f\, x := \begin{cases} \mathsf{tt} & \text{if } d\, x \downarrow \mathsf{ff} \\ \text{undef.} & \text{otherwise} \end{cases}$$

Now using EPF we obtain a code c for f and deduce that $d\, c \uparrow$ by:

$$d\, c \downarrow \mathsf{tt} \;\Leftrightarrow\; \mathsf{K}_\Theta\, c \;\Leftrightarrow\; \Theta_c\, c \downarrow \Leftrightarrow\; f\, c \downarrow \Leftrightarrow\; f\, c \downarrow \mathsf{tt} \;\Leftrightarrow\; d\, c \downarrow \mathsf{ff}$$

□

The Improved Church-Turing Proof

Theorem

Every formal system $\mathcal{S}$ weakly representing K_Θ, i.e. providing $\varphi_\mathsf{K} : \mathbb{N} \to \mathbb{S}$ with

$$\mathsf{K}_\Theta\, x \;\leftrightarrow\; \vdash \varphi_\mathsf{K}(x)$$

has an independent sentence of the form $\varphi_\mathsf{K}(c)$ for some concrete value c.

Proof.

The composition $d_\mathcal{S} \circ \varphi_\mathsf{K}$ is a candidate decider for K_Θ since:

$$\mathsf{K}_\Theta\, x \;\Leftrightarrow\; \vdash \varphi_\mathsf{K}(x) \;\Leftrightarrow\; d_\mathcal{S}\,(\varphi_\mathsf{K}(x)) \downarrow \mathsf{tt}$$

Thus there is c with $d_\mathcal{S}\,(\varphi_\mathsf{K}(c)) \uparrow$, yielding that $\varphi_\mathsf{K}(c)$ is independent. □

Only applies to sound extensions of $\mathcal{S}$...

Synthetic Recursive Inseparability

Lemma

$K^1_\Theta := \Theta_x\, x \downarrow 1$ and $K^0_\Theta := \Theta_x\, x \downarrow 0$ are recursively inseparable, in fact for every $s : \mathbb{N} \rightharpoonup \mathbb{B}$ with

$$K^1_\Theta\, x \;\to\; s\, x \downarrow \mathsf{tt} \qquad \textit{and} \qquad K^0_\Theta\, x \;\to\; s\, x \downarrow \mathsf{ff}$$

one can construct a concrete value c such that $s\, c$ diverges.

Proof.

We first define a partial function $f : \mathbb{N} \rightharpoonup \mathbb{B}$ diagonalising against s by:

$$f\, x := \begin{cases} \mathsf{tt} & \text{if } s\, x \downarrow \mathsf{ff} \\ \mathsf{ff} & \text{if } s\, x \downarrow \mathsf{tt} \\ \text{undef.} & s\, x \uparrow \end{cases}$$

Now using EPF we obtain a code c for f and deduce that $s\, c \uparrow$ by simple calculation. □

Kleene's Proof (Kleene, 1952)

Theorem

Every formal system $\mathcal{S}$ strongly separating $\mathsf{K}^1_\ominus$ and $\mathsf{K}^0_\ominus$, i.e. providing $\varphi_\mathsf{K} : \mathbb{N} \to \mathbb{S}$ with

$$\mathsf{K}^1_\ominus x \;\to\; \vdash \varphi_\mathsf{K}(x) \qquad \text{and} \qquad \mathsf{K}^0_\ominus x \;\to\; \vdash \neg\varphi_\mathsf{K}(x)$$

has an independent sentence of the form $\varphi_\mathsf{K}(c)$ for some concrete value c.

Proof.

The function $d_\mathcal{S} \circ \varphi_\mathsf{K}$ is a candidate separator for $\mathsf{K}^1_\ominus$ and $\mathsf{K}^0_\ominus$ since:

$$\begin{aligned} \mathsf{K}^1_\ominus x \;&\Rightarrow\; \vdash \varphi_\mathsf{K}(x) &&\Rightarrow\; d_\mathcal{S}\,(\varphi_\mathsf{K}(x)) \downarrow \mathsf{tt} \\ \mathsf{K}^0_\ominus x \;&\Rightarrow\; \vdash \neg\varphi_\mathsf{K}(x) &&\Rightarrow\; d_\mathcal{S}\,(\varphi_\mathsf{K}(x)) \downarrow \mathsf{ff} \end{aligned}$$

Thus there is c with $d_\mathcal{S}\,(\varphi_\mathsf{K}(c)) \uparrow$, yielding that $\varphi_\mathsf{K}(c)$ is independent. □

Immediately applies to consistent extensions of $\mathcal{S}$!

Instantiation: Essential Incompleteness of Q

To instantiate these abstract proofs to Q, we need a stronger assumption than EPF:

Axiom (CT_Q, cf. Hermes and Kirst (2022))

For every $f : \mathbb{N} \rightharpoonup \mathbb{N}$ there is a Σ_1-formula φ with: $f\,x \downarrow y \;\leftrightarrow\; Q \vdash \forall y'.\, \varphi(\overline{x}, y') \leftrightarrow y' = \overline{y}$

CT_Q implies that Q and every consistent extension of it has an independent sentence:

- CT_Q implies EPF and that Q strongly separates the respective problems $K^1_\ominus$ and $K^0_\ominus$
- Claim follows from the abstract incompleteness result

CT_Q is implied by a more conventional formulation of Church's thesis:

- EPF_μ states that every $f : \mathbb{N} \rightharpoonup \mathbb{N}$ is μ-computable (Troelstra and Van Dalen, 1988)
- Mechanised DPRM theorem (Larchey-Wendling and Forster, 2019) yields f Diophantine
- Σ_1-completeness and Rosser's trick yield that f can be captured as in CT_Q

Conclusion

Results Overview

1. Weak Church-Turing incompleteness
2. Improved Church-Turing incompleteness (using EPF)
3. Kleene's incompleteness (using EPF)
4. Essential undecidability of Q (using $\mathsf{CT_Q}$)
5. Essential undecidability of Q (using EPF_μ)

Contributions

- Translation of several incompleteness proofs into abstract and synthetic setting
  - "Synthetic computability trivialises things that should have been trivial from the beginning"
- Popularisation of Kleene's strong computational incompleteness proofs
  - Less well-known though way stronger and not much more complicated than Church-Turing
- Identification of CT_Q as suitable axiom for synthetic computability theory
  - Consistent assumption exactly factoring away Gödelisation tricks
- Coq mechanisation, systematically hyperlinked with paper
  - Only 200 lines for strongest incompleteness result, 2500 for instantiation to Q
  - Based on Coq libraries for undecidability (Forster et al., 2020) and FOL (Kirst et al., 2022)

Perspectives

- Sidestep the DPRM theorem for a less heavy-weight consistency proof of CT_Q
- Postpone/avoid the use of EPF by working against an abstract computability predicate
- Explore synthetic approaches to the second incompleteness theorem

`https://www.ps.uni-saarland.de/extras/incompleteness/`

Thanks for your attention!

Bibliography I

Bauer, A. (2006). First steps in synthetic computability theory. *Electronic Notes in Theoretical Computer Science*, 155:5–31.

Forster, Y. (2021). Church's thesis and related axioms in Coq's type theory. In Baier, C. and Goubault-Larrecq, J., editors, *29th EACSL Annual Conference on Computer Science Logic (CSL 2021)*, volume 183 of *LIPIcs*, pages 21:1–21:19, Dagstuhl, Germany.

Forster, Y., Larchey-Wendling, D., Dudenhefner, A., Heiter, E., Kirst, D., Kunze, F., Smolka, G., Spies, S., Wehr, D., and Wuttke, M. (2020). A Coq library of undecidable problems. In *CoqPL 2020*, New Orleans, LA, United States.

Gödel, K. (1931). Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für mathematik und physik*, 38(1):173–198.

Hermes, M. and Kirst, D. (2022). An analysis of Tennenbaum's theorem in constructive type theory. In *7th International Conference on Formal Structures for Computation and Deduction (FSCD 2022)*.

Kirst, D. and Hermes, M. (2021). Synthetic undecidability and incompleteness of first-order axiom systems in Coq. In *12th International Conference on Interactive Theorem Proving (ITP 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Bibliography II

Kirst, D., Hostert, J., Dudenhefner, A., Forster, Y., Hermes, M., Koch, M., Larchey-Wendling, D., Mück, N., Peters, B., Smolka, G., and Wehr, D. (2022). A Coq library for mechanised first-order logic. In *The Coq Workshop*.

Kleene, S. C. (1951). A symmetric form of Gödel's theorem. *Journal of Symbolic Logic*, 16(2).

Kleene, S. C. (1952). *Introduction to Metamathematics*.

Kreisel, G. (1970). Church's thesis: a kind of reducibility axiom for constructive mathematics. In *Studies in Logic and the Foundations of Mathematics*, volume 60, pages 121–150.

Larchey-Wendling, D. and Forster, Y. (2019). Hilbert's tenth problem in Coq. In *4th International Conference on Formal Structures for Computation and Deduction*, volume 131 of *LIPIcs*, pages 27:1–27:20.

O'Connor, R. (2005). Essential incompleteness of arithmetic verified by Coq. In *International Conference on Theorem Proving in Higher Order Logics*, pages 245–260. Springer.

Paulson, L. C. (2015). A mechanised proof of Gödel's incompleteness theorems using Nominal Isabelle. *Journal of Automated Reasoning*, 55(1):1–37.

Bibliography III

Popescu, A. and Traytel, D. (2019). A formally verified abstract account of Gödel's incompleteness theorems. In *International Conference on Automated Deduction*, pages 442–461. Springer.

Richman, F. (1983). Church's thesis without tears. *The Journal of symbolic logic*, 48(3):797–803.

Rosser, B. (1936). Extensions of some theorems of Gödel and Church. *The journal of symbolic logic*, 1(3):87–91.

Shankar, N. (1986). *Proof-checking metamathematics*. The University of Texas at Austin. PhD Thesis.

Swan, A. W. and Uemura, T. (2019). On Church's thesis in cubical assemblies. *Mathematical Structures in Computer Science*, pages 1–20.

Troelstra, A. S. and Van Dalen, D. (1988). *Constructivism in Mathematics*. Vol. 121 of Studies in Logic and the Foundations of Mathematics. North-Holland, Amsterdam.

Mechanised Incompleteness

Shankar (1986)

First full mechanisation of G1
in the Boyer-Moore theorem prover

O'Connor (2005)

Constructive mechanisation of G1
in the Coq proof assistant

Paulson (2015)

Mechanisation of G1 and G2
in Isabelle/HOL

Popescu and Traytel (2019)

Abstract preconditions for G1 and G2
in Isabelle/HOL

Kirst and Hermes (2021)

Weak G1 via synthetic undecidability
in the Coq proof assistant