Very Weak, Essentially Undecidable Set Theories

Domenico Cantone, Eugenio Omodeo, and
Mattia Panettiere

CILC 2021

September 7, 2021

Δ_0-formulae

Definition

A Δ_0-formula is a set-theoretic formula where all the quantifiers are bounded, i.e., are of the form:

$$\begin{array}{lcl} (\forall x \in y)\varphi & \equiv & \forall x(x \in y \rightarrow \varphi) \\ (\exists x \in y)\varphi & \equiv & \exists x(x \in y \wedge \varphi) \end{array}$$

Definition

A Δ_0-formula is of complexity n if it can be rewritten through a Tarski-Mostowski computation in prenex form with $n-1$ quantifier alternations, starting with a universal quantifier.

(Essential) Undecidability

Definition

A set theory Θ is said to be UNDECIDABLE w.r.t. a class C of formulae if the satisfiability problem is unsolvable, i.e., if given a formula $\varphi \in C$, there is no algorithm that finds if there is a set assignment such that the formula holds.

Definition

A set theory Θ is ESSENTIALLY UNDECIDABLE if every one of its consistent recursively axiomatizable extensions is undecidable.

Gödel arguments for essential undecidability

Gödel proved essential incompleteness for theories expressing arithmetic.

Incompleteness and decidability are unrelated in general, but:

Gödel arguments for essential undecidability

Gödel proved essential incompleteness for theories expressing arithmetic.

Incompleteness and decidability are unrelated in general, but:

- we consider existential closures of classes C of Δ_0 formulae;
- a YES answer to the decision problem for φ implies $T \vdash \varphi^{\exists}$;
- a NO answer to the decision problem for φ implies $T \vdash \neg(\varphi^{\exists})$;

Gödel arguments for essential undecidability

Gödel proved essential incompleteness for theories expressing arithmetic.

Incompleteness and decidability are unrelated in general, but:

- we consider existential closures of classes C of Δ_0 formulae;
- a $\mathrm{Y\scriptstyle ES}$ answer to the decision problem for φ implies $T \vdash \varphi^{\exists}$;
- a $\mathrm{N\scriptstyle O}$ answer to the decision problem for φ implies $T \vdash \neg(\varphi^{\exists})$;

Hence, in a consistent set theory T:

$$\text{Decidability wrt } C \implies \text{Completeness wrt } C$$

Our core theory

- Empty Set $\exists x \forall y \in x \; \neg y \in x$
- Adjunction $\forall x \forall y \exists z \forall w \big(w \in z \leftrightarrow (w = y \;\vee\; w \in x) \big)$
- Removal $\forall x \forall y \exists z \forall w \big(w \in z \leftrightarrow (\neg w = y \;\wedge\; w \in x) \big)$
- Regularity $\forall x \exists m \forall y \big(y \in x \rightarrow (m \in x \;\wedge\; \neg y \in m) \big)$

Our core theory

- Empty Set $\exists x \forall y \in x \ \neg y \in x$
- Adjunction $\forall x \forall y \exists z \forall w \big(w \in z \leftrightarrow (w = y \ \vee \ w \in x)\big)$
- Removal $\forall x \forall y \exists z \forall w \big(w \in z \leftrightarrow (\neg w = y \ \wedge \ w \in x)\big)$
- Regularity $\forall x \exists m \forall y \big(y \in x \rightarrow (m \in x \ \wedge \ \neg y \in m)\big)$

We will also consider two extensions with:

- Separation for any φ, $\forall u \exists s \forall v \big(v \in s \leftrightarrow (v \in u \ \wedge \ \varphi)\big)$
- Finitude $\forall f (\forall t \in f)(\exists a \in f)(\forall b \in f)\big((\forall d \in b) d \in a \rightarrow b = a\big)$

Essential Undecidability in very weak foundational theories

- We already had essential undecidability w.r.t. $(\forall\exists\forall)_0$-formulae.
- We have shown Gödel arguments w.r.t. $(\forall\exists)_0$-formulae!

Essential Undecidability in very weak foundational theories

- We already had essential undecidability w.r.t. $(\forall\exists\forall)_0$-formulae.
- We have shown Gödel arguments w.r.t. $(\forall\exists)_0$-formulae!

Objectives:

- Express naturals and strongly represent total recursive functions;
- Find an encoding for formulae with a $(\forall\exists)_0$-definable total order;
- Define a $\mathsf{Proof}(x, y)$ predicate;
- Prove an analogous of the Fixpoint Theorem;
- Proceed with the standard Gödel arguments.

Pairs

Consider the classical pairing function $\langle x, y \rangle = \{x, \{x, y\}\}$.

Pairs

Consider the classical pairing function $\langle x, y \rangle = \{x, \{x, y\}\}$.

$$y = \pi_2(p) \quad \overset{\text{Def}}{\longleftrightarrow} \quad (\exists x \in p)(\exists q \in p)\Big(\; x \in q \;\wedge\; y \in q \;\wedge \\ (\forall z \in q)(x = z \;\vee\; y = z) \;\Big)$$

Pairs

Consider the classical pairing function $\langle x, y \rangle = \{x, \{x, y\}\}$.

$$y = \pi_2(p) \quad \overset{\text{Def}}{\longleftrightarrow} \quad (\exists x \in p)(\exists q \in p)\Big(\quad x \in q \ \wedge \ y \in q \ \wedge \\ (\forall z \in q)(x = z \ \vee \ y = z) \quad \Big)$$

$$(\exists\forall)_0$$

Pairs - 2

We opted for:

$$\begin{aligned} y @ x &:= \{ y \ \texttt{less} \ x \, , \ y \ \texttt{with} \ x \} \, , \\ \langle x, y \rangle &:= (x @ y) @ x \, . \end{aligned}$$

- Well suited for our axiomatic system;
- Projection extraction requires only existential quantifiers;
- No particular cases or exceptions;
- No pair is an ordinal.

Naturals with Axiom of Specification

Under the core with an instance of specification, we proved:

$$\begin{aligned}\mathrm{Num}(X) \quad \longleftrightarrow \quad & \forall\, y \in X^{+} \left(y = \varnothing \vee \exists\, z \in X \;\; z^{+} = y \right) \qquad \wedge \\ & (\forall\, u, v \in X \texttt{ less } \varnothing)\left(u \in v \vee v \in u \vee v = u \right),\end{aligned}$$

Under the core with an instance of specification, we proved:

$$\mathrm{Num}(X) \;\longleftrightarrow\; \forall\, y \in X^{+} \left(y = \varnothing \vee \exists\, z \in X \;\; z^{+} = y \right) \quad \wedge$$
$$(\forall\, u, v \in X \;\texttt{less}\; \varnothing)\,(\, u \in v \vee v \in u \vee v = u \,)\,,$$

$$(\forall\exists\forall)_0\,, \quad \text{as} \quad t = s^{+} \xleftrightarrow{\text{Def}} (\forall x \in t)(x = s \;\vee\; x \in s).$$

Naturals with Axiom of Specification

$$
\begin{aligned}
\mathsf{Num}(X) \overset{\text{Def}}{\longleftrightarrow}\ & \mathsf{Quadruple}(X) \wedge \\
& \mathsf{Fun}(\pi_4(X)) \wedge \pi_1(X) = \pi_2(X)^- \wedge \pi_3(X) = \pi_2(X)^+ \wedge \\
& (\forall n \in \pi_2(X))\mathsf{Triple}(\pi_4(X)(n)) \wedge \operatorname{dom} \pi_4(X) = \pi_2(X) \wedge \\
& (\forall t \in \operatorname{ran} \pi_4(X))(\pi_1(t) = \pi_2(t)^- \wedge \pi_3(t) = \pi_2(t)^+) \wedge \\
& (\forall u, v \in \pi_2(X) \mathtt{\ less\ } \varnothing)(u \in v \vee v \in u \vee u = v) \wedge \\
& (\forall y \in \pi_3(X))\big(y = \varnothing \vee (\exists z \in \pi_2(X))\pi_3((\pi_4(X))(z)) = y\big)
\end{aligned}
$$

$$
\langle n^-, n, n^+, m \mapsto \langle m^-, m, m^+\rangle \quad \text{for} \quad m \in n\rangle
$$

Naturals with Axiom of Specification

$$
\begin{aligned}
\mathsf{Num}(X) \overset{\mathrm{Def}}{\longleftrightarrow}\ & \mathsf{Quadruple}(X) \wedge \\
& \mathsf{Fun}(\pi_4(X)) \ \wedge\ \pi_1(X) = \pi_2(X)^- \ \wedge\ \pi_3(X) = \pi_2(X)^+ \ \wedge \\
& (\forall n \in \pi_2(X))\mathsf{Triple}(\pi_4(X)(n)) \ \wedge\ \mathrm{dom}\, \pi_4(X) = \pi_2(X) \ \wedge \\
& (\forall t \in \mathrm{ran}\, \pi_4(X))(\pi_1(t) = \pi_2(t)^- \ \wedge\ \pi_3(t) = \pi_2(t)^+) \ \wedge \\
& (\forall u, v \in \pi_2(X) \,\mathtt{less}\, \varnothing)(u \in v \ \vee\ v \in u \ \vee\ u = v) \ \wedge \\
& (\forall y \in \pi_3(X))\big(y = \varnothing \ \vee\ (\exists z \in \pi_2(X))\pi_3((\pi_4(X))(z)) = y\big)
\end{aligned}
$$

$$
\langle n^-, n, n^+, m \mapsto \langle m^-, m, m^+ \rangle \quad \text{for} \quad m \in n \rangle
$$

Naturals with Axiom of Specification

$$\begin{aligned}
\mathsf{Num}(X) \overset{\text{Def}}{\longleftrightarrow}\ & \mathsf{Quadruple}(X) \wedge \\
& \mathsf{Fun}(\pi_4(X)) \ \wedge\ \pi_1(X) = \pi_2(X)^- \ \wedge\ \pi_3(X) = \pi_2(X)^+ \ \wedge \\
& (\forall n \in \pi_2(X))\mathsf{Triple}(\pi_4(X)(n)) \ \wedge\ \mathsf{dom}\, \pi_4(X) = \pi_2(X) \ \wedge \\
& (\forall t \in \mathsf{ran}\, \pi_4(X))(\pi_1(t) = \pi_2(t)^- \ \wedge\ \pi_3(t) = \pi_2(t)^+) \ \wedge \\
& (\forall u, v \in \pi_2(X) \ \mathtt{less}\ \varnothing)(u \in v \ \vee\ v \in u \ \vee\ u = v) \ \wedge \\
& (\forall y \in \pi_3(X))\big(y = \varnothing \ \vee\ (\exists z \in \pi_2(X))\pi_3((\pi_4(X))(z)) = y\big)
\end{aligned}$$

$$\langle n^-, n, n^+, m \mapsto \langle m^-, m, m^+\rangle \quad \text{for} \quad m \in n\rangle$$

Naturals with Axiom of Specification

$$\begin{aligned}\mathsf{Num}(X) \overset{\text{Def}}{\longleftrightarrow}\ & \mathsf{Quadruple}(X) \wedge \\ & \mathsf{Fun}(\pi_4(X)) \ \wedge\ \pi_1(X) = \pi_2(X)^- \ \wedge\ \pi_3(X) = \pi_2(X)^+ \wedge \\ & (\forall n \in \pi_2(X))\mathsf{Triple}(\pi_4(X)(n)) \ \wedge\ \operatorname{dom} \pi_4(X) = \pi_2(X) \wedge \\ & (\forall t \in \operatorname{ran} \pi_4(X))(\pi_1(t) = \pi_2(t)^- \ \wedge\ \pi_3(t) = \pi_2(t)^+) \wedge \\ & (\forall u, v \in \pi_2(X) \,\texttt{less}\, \varnothing)(u \in v \ \vee\ v \in u \ \vee\ u = v) \wedge \\ & (\forall y \in \pi_3(X))\big(y = \varnothing \ \vee\ (\exists z \in \pi_2(X))\pi_3((\pi_4(X))(z)) = y\big)\end{aligned}$$

$$\langle n^-, n, n^+, m \mapsto \langle m^-, m, m^+\rangle \quad \text{for} \quad m \in n\rangle$$

Naturals with Axiom of Specification

$$\begin{aligned}
&\mathsf{Num}(X) \overset{\text{Def}}{\longleftrightarrow} \mathsf{Quadruple}(X) \wedge \\
&\quad \mathsf{Fun}(\pi_4(X)) \;\wedge\; \pi_1(X) = \pi_2(X)^- \;\wedge\; \pi_3(X) = \pi_2(X)^+ \;\wedge \\
&\quad (\forall n \in \pi_2(X))\mathsf{Triple}(\pi_4(X)(n)) \;\wedge\; \mathsf{dom}\,\pi_4(X) = \pi_2(X) \;\wedge \\
&\quad (\forall t \in \mathsf{ran}\,\pi_4(X))(\pi_1(t) = \pi_2(t)^- \;\wedge\; \pi_3(t) = \pi_2(t)^+) \;\wedge \\
&\quad (\forall u, v \in \pi_2(X) \;\mathtt{less}\; \varnothing)(u \in v \;\vee\; v \in u \;\vee\; u = v) \;\wedge \\
&\quad (\forall y \in \pi_3(X))\big(y = \varnothing \;\vee\; (\exists z \in \pi_2(X))\pi_3((\pi_4(X))(z)) = y\big)
\end{aligned}$$

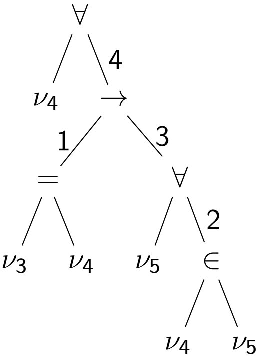
$$\langle n^-, n, n^+, m \mapsto \langle m^-, m, m^+\rangle \quad \text{for} \quad m \in n\rangle$$

$$\begin{aligned}
&(\forall\, u, v \in X \;\mathtt{less}\; \varnothing)\,(\,u \in v \vee v \in u \vee v = u\,) \wedge \\
&\forall\, y \in X^+ \,\big(\,y = \varnothing \vee \exists\, z \in X \quad z^+ = y\,\big)
\end{aligned}$$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \rightarrow \forall \nu_5 (\nu_4 \in \nu_5) \big)$$



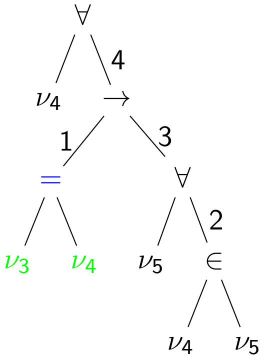
Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$$\begin{array}{lll} 0 & \mapsto & 7 \\ 1 & \mapsto & \langle =, 3, 4 \rangle \\ 2 & \mapsto & \langle \in, 4, 5 \rangle \\ 3 & \mapsto & \langle \forall, 5, 2 \rangle \\ 4 & \mapsto & \langle \rightarrow, 1, 3 \rangle \\ 5 & \mapsto & \langle \forall, 4, 4 \rangle \end{array}$$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \to \forall \nu_5 (\nu_4 \in \nu_5) \big)$$



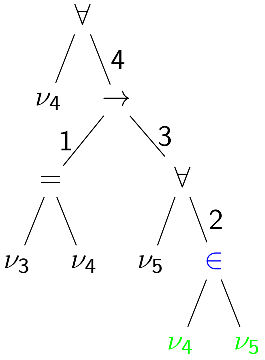
Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$$\begin{array}{lll} 0 & \mapsto & 7 \\ 1 & \mapsto & \langle =, 3, 4 \rangle \\ 2 & \mapsto & \langle \in, 4, 5 \rangle \\ 3 & \mapsto & \langle \forall, 5, 2 \rangle \\ 4 & \mapsto & \langle \to, 1, 3 \rangle \\ 5 & \mapsto & \langle \forall, 4, 4 \rangle \end{array}$$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \rightarrow \forall \nu_5 (\nu_4 \in \nu_5) \big)$$



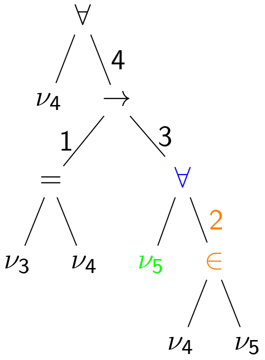
Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$$\begin{array}{lcl} 0 & \mapsto & 7 \\ 1 & \mapsto & \langle =, 3, 4 \rangle \\ 2 & \mapsto & \langle \in, 4, 5 \rangle \\ 3 & \mapsto & \langle \forall, 5, 2 \rangle \\ 4 & \mapsto & \langle \rightarrow, 1, 3 \rangle \\ 5 & \mapsto & \langle \forall, 4, 4 \rangle \end{array}$$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \rightarrow \forall \nu_5 (\nu_4 \in \nu_5) \big)$$



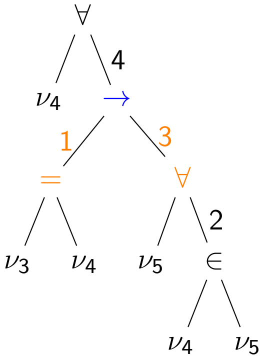
Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$$\begin{array}{lcl} 0 & \mapsto & 7 \\ 1 & \mapsto & \langle =, 3, 4 \rangle \\ 2 & \mapsto & \langle \in, 4, 5 \rangle \\ 3 & \mapsto & \langle \forall, 5, 2 \rangle \\ 4 & \mapsto & \langle \rightarrow, 1, 3 \rangle \\ 5 & \mapsto & \langle \forall, 4, 4 \rangle \end{array}$$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \rightarrow \forall \nu_5 (\nu_4 \in \nu_5) \big)$$



Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$0 \mapsto 7$

$1 \mapsto \langle =, 3, 4 \rangle$

$2 \mapsto \langle \in, 4, 5 \rangle$

$3 \mapsto \langle \forall, 5, 2 \rangle$

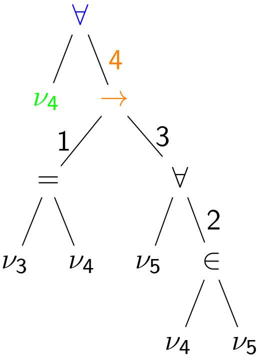
$4 \mapsto \langle \rightarrow, 1, 3 \rangle$

$5 \mapsto \langle \forall, 4, 4 \rangle$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \rightarrow \forall \nu_5 (\nu_4 \in \nu_5)\big)$$



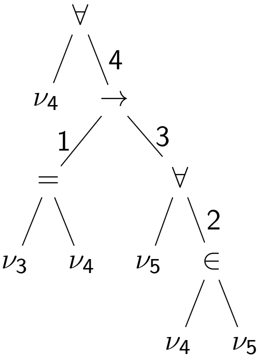
Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$$\begin{array}{lll} 0 & \mapsto & 7 \\ 1 & \mapsto & \langle =, 3, 4 \rangle \\ 2 & \mapsto & \langle \in, 4, 5 \rangle \\ 3 & \mapsto & \langle \forall, 5, 2 \rangle \\ 4 & \mapsto & \langle \rightarrow, 1, 3 \rangle \\ 5 & \mapsto & \langle \forall, 4, 4 \rangle \end{array}$$

Codes

Take a set formula, e.g.:

$$\forall \nu_4 \big(\nu_3 = \nu_4 \to \forall \nu_5(\nu_4 \in \nu_5)\big)$$



Function from $\{0, 1, 2, 3, 4, 5\}$ s.t.:

$$\begin{array}{lcl} 0 & \mapsto & 7 \\ 1 & \mapsto & \langle =, 3, 4 \rangle \\ 2 & \mapsto & \langle \in, 4, 5 \rangle \\ 3 & \mapsto & \langle \forall, 5, 2 \rangle \\ 4 & \mapsto & \langle \to, 1, 3 \rangle \\ 5 & \mapsto & \langle \forall, 4, 4 \rangle \end{array}$$

$\mathsf{Cod}(x)$, $\leq_C$, and Next_C are $(\forall\exists)_0$.
With adjustments, so is $\mathsf{Form}(x)$.

Proofs

Several problems:

- Check if a variable is bound $\rightsquigarrow$ BoundList predicate;
- Check if two codes are equivalent $\rightsquigarrow$ CLCopy and CRCopy,
  - e.g. to check if modus ponens is applicable;
- Check if a formula is obtained from another through renaming;
- Do all of this with a $(\forall\exists)_0$ formula!

Proofs

Several problems:

- Check if a variable is bound $\rightsquigarrow$ BoundList predicate;
- Check if two codes are equivalent $\rightsquigarrow$ CLCopy and CRCopy,
  - e.g. to check if modus ponens is applicable;
- Check if a formula is obtained from another through renaming;
- Do all of this with a $(\forall\exists)_0$ formula!

Again, we applied the technique of using functions/tuples to store in an easily accessible way complex information.

Proofs

Proofs are tuples containing:

- A list of all the subformulae of the formulae in the proof,
  - seen as triples containing clean left and right copies;
- A list of indices pointing to the first list (the proof sequence);
- A list of bound list for each subformula;
- A list of the domains of the codes of all subformulae.

Proofs

Proofs are tuples containing:

- A list of all the subformulae of the formulae in the proof,
  - seen as triples containing clean left and right copies;
- A list of indices pointing to the first list (the proof sequence);
- A list of bound list for each subformula;
- A list of the domains of the codes of all subformulae.

It is possible to exploit the several parts to characterize all the rules, rename resolution, and the axioms.

By some technical considerations, this allows to prove the desired result.

Essential Undecidability - Novel results

Theorem

For every extension T' of T where natural numbers are sufficiently expressible through a $(\forall\exists)_0$ formula, i.e., every total recursive function on naturals is strongly representable, any Cod$-$*consistent recursively axiomatizable extension Θ of T' is undecidable with respect to $(\forall\exists)_0$ formulae.*

Corollary

T plus a single instance of the axiom schema of separation is essentially undecidable with respect to $(\forall\exists)_0$ formulae.

Corollary

T plus an axiom forcing the universe to be the one of hereditarily finite sets is essentially undecidable with respect to $(\forall\exists)_0$ formulae.

Some Future Developments

- Further refine the results applying similar techniques:
  - Find some minimal essentially undecidable theories wrt $(\forall\exists)_0$.
  - Finitude is restrictive.
  - Maybe the axiom of separation can be dropped.
- Generalize the techniques and tighten the class of formulae.
- Try to use a $(\forall\exists)_0$ characterization of the axiom of inifinity as a base for complexity reduction.

Thank you for your attention!

Thank you for your attention!

Questions?