Conservativity of weak Kőnig's lemma (a proof from the book)

Jeremy Avigad

Department of Philosophy
Department of Mathematical Sciences

Charles C. Hoskinson Center
for Formal Mathematics

Carnegie Mellon University

September 2021

Overview

Settings

Quantifier-free arithmetic
where we can say things about any number but not all of them

First-order arithmetic
where we can say things about all the numbers

Second-order arithmetic
where we can talk about sets of numbers too

Overview

Cast of Characters

Primitive recursive arithmetic
an explication of "finitistic" reasoning à la Hilbert and Bernays

$I\Sigma_1$
a fragment of first-order arithmetic based on Σ_1 induction

RCA_0
a fragment of second-order arithmetic for reasoning about computable sets

WKL_0
a fragment of second-order arithmetic with an additional compactness principle

Overview

In the early 1920s, Hilbert proposed a means of securing modern, infinitary methods: prove consistency using only a safe, finitistic part.

Gödel's second incompleteness theorem rules that out.

A modified Hilbert's program: interpret as much mathematics as you can in theories that are more meager than full-blown set theory.

Characterize those theories in terms of their "concrete" combinatorial and computational consequences.

Overview

Twentieth century proof theory showed:

- We can interpret a lot of mathematics in theories that are not very strong.
- We can characterize their computational and combinatorial strength in various ways.

The latter has informed developments in computer science and mathematics.

We have subsystems of second-order arithmetic, RCA_0, WKL_0, ACA_0, ATR_0, and Π^1_1-CA_0, and reverse mathematics.

We have double-negation translations, ordinal analysis, realizability and functional interpretation, and more.

Theorems

Theorem (Parsons, Mints, Takeuti). $\mathsf{I}\Sigma_1$ is Π_2 conservative over PRA.

Theorem (Ignatovic, Solovay). There is an iterated exponential *speedup* in the previous results.

Theorem. RCA_0 is interpretable in $\mathsf{I}\Sigma_1$ and hence Π^1_1 conservative over it without speedup.

Theorems

Theorem (Friedman). WKL_0 is Π_2 conservative over PRA.

Theorem (Harrington). WKL_0 is Π^1_1 conservative over RCA_0.

Theorem (Hájek, Avigad). WKL_0 is interpretable in RCA_0, so there is no speedup in the previous result.

Interesting mathematics can be carried out in WKL_0 and related theories.

The results say something about the logical strength of common patterns of mathematical reasoning.

The elimination of compactness is central to Kohlenbach's *Proof Mining* program.

Table of contents

A personal note

The fact that there is no speedup between WKL_0 and RCA_0 was one of my first published results.

Now, many years later, I am about to send a textbook called *Mathematical Logic* to the Cambridge University Press.

It contains a streamlined proof of that result.

Primitive recursive arithmetic

The set of primitive recursive functions includes

- zero, 0
- successor, $S(x) = x + 1$
- projections, $p_i^n(x_1, \ldots, x_n) = x_i$

and are closed under

- composition: $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_n(\vec{x}))$
- primitive recursion:

$$f(0, \vec{z}) = g(\vec{z}), \quad f(x + 1, \vec{z}) = h(f(x, \vec{z}), x, \vec{z})$$

Primitive recursive arithmetic

Primitive recursive arithmetic is an axiomatic theory, with

- $0 \neq S(x)$, $S(x) = S(y) \to x = y$
- defining equations for the primitive recursive functions
- quantifier-free induction:

$$\frac{A(0) \qquad A(x) \to A(x+1)}{A(t)}$$

Variables range over arbitrary numbers, but we can't quantify over them explicitly.

Primitive recursive arithmetic

All reasonable computable functions are primitive recursive, and all reasonable facts about them can be proved in PRA.

(I hope the book makes this vague claim compelling.)

PRA can also be presented as a first-order theory (classical or intuitionistic). Herbrand's theorem tells us this a conservative extension.

It is surprisingly hard to find ordinary mathematical theorems that can be stated in the language of PRA but not proved there.

So we can think of PRA as a robust theory for reasoning about finite objects.

First-order arithmetic

First-order arithmetic is essentially PRA plus induction. *Peano arithmetic* (PA) is classical, *Heyting arithmetic* (HA) is intuitionistic.

Language: $0, S, +, \times, <$.

Axioms: quantifier-free defining axioms, induction.

A formula is

- Δ_0 if every quantifier is bounded, $\forall x < t\ A$ or $\exists x < t\ A$.
- Σ_1 if of the form $\exists \vec{x}\ A$, $A \in \Delta_0$
- Π_1 if of the form $\forall \vec{x}\ A$, $A \in \Delta_0$
- Δ_1 if equivalent to Σ_1 and Π_1

Primitive recursive functions and relations have Δ_1 definitions.

Conservativity of $I\Sigma_1$ over PRA

$I\Sigma_1$ is the restriction of PA with induction for only Σ_1 formulas.

This theory suffices to define the primitive recursive functions, and hence interpret PRA. Conversely:

Theorem (Parsons, Mints, Takeuti). $I\Sigma_1$ is conservative over PRA for Π_2 sentences: if

$$I\Sigma_1 \vdash \forall x\, \exists y\, A(x, y),$$

with A is Δ_0, then

$$\mathrm{PRA} \vdash A(x, f(x))$$

for some function symbol f.

Conservativity of $I\Sigma_1$ over PRA

There are various ways to prove this theorem.

Syntactic proofs:

- Using cut elimination or normalization.
- Using the Dialectica interpretation (plus normalization).

Model-theoretic proofs:

- A model-theoretic argument due to Friedman.
- Another one in the book.

Speedup of $I\Sigma_1$ over PRA

Theorem (Ignatovic). There are a polynomial $p(n)$ and a sequence of quantifier-free formulas $A_0, A_1, \ldots, A_{n-1}$ such that for every n:

- There is a proof of A_n in $I\Sigma_1$ of length $p(n)$.
- The smallest proof of A_n in PRA has length 2^0_n.

Here 2^k_n is the iterated exponential, $2^k_0 = k$, $2^k_{n+1} = 2^{2^k_n}$.

The proof uses Solovay's method of *shortening of cuts* to construct efficient consistency proofs of PRA_n in $I\Sigma_1$.

Speedup of $I\Sigma_1$ over PRA

John Burgess has pointed out, correctly, that this speaks against thinking of the result as a finitistic reduction.

Responses:

- Accept finitism plus a reflection principle.
- Accept finitism with higher types.
- Give up Σ_1 induction.
- In practice, the speedup isn't nearly so bad.
- We still get primitive recursive bounds.

Second order arithmetic

The language is two-sorted:

- variables $x, y, z, \ldots$ and functions $0, S, +, \times$ on one sort
- variables $X, Y, Z, \ldots$ on the other sort
- a relation $t \in X$ between the two sorts

Axioms:

- axioms of PA, with induction extended to the bigger language
- comprehension: $\exists X \, \forall y \, (y \in X \leftrightarrow A(y, \vec{z}))$

The "standard model" is $(\mathbb{N}, \mathcal{P}(\mathbb{N}), \ldots)$, but there are smaller ones.

An ω-model is a model where the first-order part is standard, i.e. $\mathbb{N}$.

Second order arithmetic

From a proof-theoretic perspective, second-order arithmetic is very strong.

We obtain weaker systems by:

- restricting comprehension
- restricting induction

Subsystems of second-order arithmetic

The big five:

- RCA_0: recursive (Δ^0_1) comprehension
  (formalized computable analysis)
- WKL_0: weak König's lemma
  (a form of compactness)
- ACA_0: arithmetic comprehension
  (analytic principles like the least-upper bound principle.)
- ATR_0: transfinitely iterated arithmetic comprehension
  (transfinite constructions)
- $\Pi^1_1\text{–}\mathsf{CA}_0$: Π^1_1 comprehension
  (strong analytic principles)

We will focus on the first two.

RCA_0

The axioms of RCA_0 are as follows:

- quantifier-free axioms for $0, S, +, \times, <$
- induction, restricted to Σ_1 formulas (with both number and set parameters):

$$A(0) \wedge \forall x\, (A(x) \to A(x+1)) \to \forall x\, A(x)$$

- the recursive comprehension axiom, (RCA):

$$\forall x\, (A(x) \leftrightarrow B(x)) \to \exists Y\, \forall x\, (x \in Y \leftrightarrow A(x))$$

  where A is Σ_1 and B is Π_1.

RCA_0

Notice that the induction schema includes set induction:

$$0 \in Y \wedge \forall x\, (x \in Y \to x + 1 \in Y) \to \forall x\, (x \in Y).$$

It is slightly stronger.

Since RCA_0 includes $\mathrm{I}\Sigma_1$, we can act as though primitive recursive arithmetic is "built-in."

RCA_0

(RCA) says that a set exists if it has a computably enumerable definition as well as a co-computably enumerable definition (relative to others sets in the universe).

Roughly, it allows you to define computable sets and relations.

Let REC denote the set of recursive sets. Then $(\mathbb{N}, \mathrm{REC}, \ldots)$ is the minimal ω-model.

Analysis in RCA_0 is roughly "formalized computable analysis."

It is straightforward to interpret RCA_0 in $\mathsf{I}\Sigma_1$, by interpreting the set variables as ranging over computer programs.

Synopsis

Where we are:

- PRA is a robust theory of finitistic reasoning.
- $\mathsf{I}\Sigma_1$ is Π_2 conservative over PRA, but with speedup.
- RCA_0 is Π^1_1 conservative over $\mathsf{I}\Sigma_1$, with no speedup.

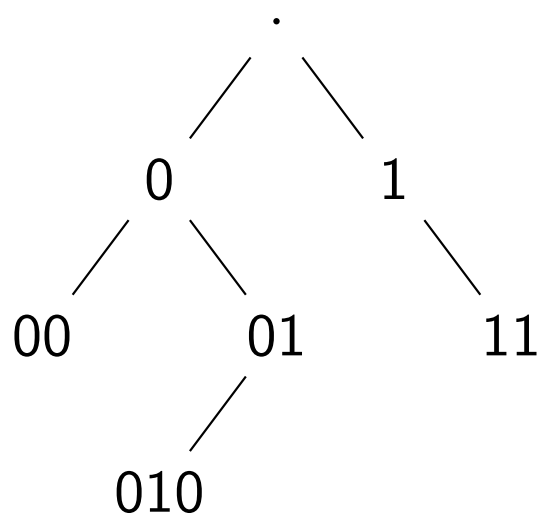
What's left:

- I'll tell you about WKL_0.
- I'll tell you why it is Π^1_1 conservative over RCA_0, without speedup.

Binary trees

If σ and τ are finite binary sequences, write $\sigma \subseteq \tau$ to mean that σ is an initial segment of τ.

A *binary tree* is a set of finite binary sequences closed under initial segments.



In a formal theory of arithmetic, finite binary sequences can be represented by numbers.

Binary trees

Binary trees can be infinite.

A *path* through a binary tree is a maximal set of compatible nodes.

An infinite path corresponds to an infinite binary sequence, which corresponds to a set of natural numbers.

Every infinite binary tree has an infinite path: just step through the tree maintaining the property that there are infinitely many descendents.

This is *Kőnig's lemma* for binary trees.

WKL_0

We can express these in the language of second-order arithmetic:

$$\mathsf{Tree}(T) \equiv \forall \sigma, \tau \, (\tau \in T \wedge \sigma \subseteq \tau \to \sigma \in T).$$

We can say T is infinite as follows:

$$\mathsf{Infinite}(T) \equiv \forall n \, \exists \sigma \, (\sigma \in T \wedge \mathsf{length}(\sigma) = n).$$

Write $\sigma \subset P$ for

$$\forall i < \mathsf{length}(\sigma) \, (i \in P \leftrightarrow (\sigma)_i = 1).$$

Then define

$$\mathsf{Path}(P, T) \equiv \forall \sigma \subset P \, (\sigma \in T).$$

WKL_0

Weak König's lemma (WKL) says that every infinite binary tree has a path:

$$\forall T\ (\mathsf{Tree}(T) \wedge \mathsf{Infinite}(T) \to \exists P\ \mathsf{Path}(P, T)).$$

The theory WKL_0 is $\mathsf{RCA}_0 + (\mathsf{WKL})$.

We will see that there are computable trees with no computable path. So RCA_0 doesn't prove (WKL).

Theorem (Harrington). WKL_0 is Π^1_1 conservative over RCA_0.

WKL_0

Over RCA_0, (WKL) is equivalent to each of these:

- the Heine-Borel theorem (for $[0, 1]$)
- Every open cover of $\{0, 1\}^\omega$ has a finite subcover.
- Every continuous function on $[0, 1]$ is uniformly continuous
- Every continuous function on $[0, 1]$ is bounded

Here, $[0, 1]$ can be replaced by any compact space.

Table of contents

Conservativity of WKL_0

Harrington's original argument was a *forcing argument*.

Start with a model $\mathcal{M}$ of RCA_0. Suppose T is an infinite binary tree in the model with no infinite path.

We want to add a set, X, that is a path through T, and all the sets computable from it. Call that $\mathcal{M}[X]$.

That can break some axioms, in particular, Σ_1 induction.

The idea: add a *generic* path through T, to preserve Σ_1 induction.

Conservativity of WKL_0

Dolly Parton: "It costs me a lot of money to look this cheap."

It takes a lot of work to add an X that is bland and innocuous.

Build it as a limit of nodes of T, $\sigma_0 \subseteq \sigma_1 \subseteq \sigma_2 \cdots$.

Make sure that the only properties that it has are those it is *forced* to have at some finite stage of the construction.

In particular, Σ_1 properties of the generic set can be described in the original model.

This adds one path through a tree. Now iterate.

Computable binary trees

If we represent binary sequences as natural numbers, a binary tree is just a set of numbers.

Theorem (Kleene). There is a computable infinite binary tree T with no computable path.

In other words, T is a computable set, but no path P through T is a computable set.

Definition. A set is *A computably enumerable* (c.e.) if it is the range of a computable function, $A = \{\varphi_e(0), \varphi_e(1), \varphi_e(2), \ldots\}$.

A computably enumerable set is definable by a Σ_1 formula, and vice versa.

Computable binary trees

Theorem. There are disjoint computably enumerable sets A and B that are *computably inseparable*, i.e. there is no computable set C such that $A \subseteq C$ and $B \subseteq \overline{C}$.

Proof. The sets $A = \{n \mid \varphi_n(n) \downarrow = 1\}$ and $B = \{n \mid \varphi_n(n) \downarrow = 0\}$ will do.

Build the Kleene tree T as follows: put σ in T if and only if, running Turing machines at most length(σ) steps, it is consistent that σ is the initial segment of a separation of A from B.

Then any path P through T will *be* a separation of A from B, and hence not computable.

Computable binary trees

In fact, there is a close connection between paths through computable trees and separations of computably enumerable sets.

Given disjoint c.e. sets A and B, there is a computable infinite binary tree T such that a separation of A from B can be computed from any path through T.

Conversely, given a computable infinite binary tree, T, there are disjoint c.e. sets A and B such that a path through T can be computed from any separation of A from B.

Everything I have said relativizes, i.e. we can replace "computable" by "computable in X" and "computably enumerable" by "computably enumerable in X."

The low basis theorem

If X is any set, X' denotes the *Turing jump* of X, that is, the halting problem relative to X.

It is a complete computably enumerable (Σ_1-definable) set.

Saying that there are only finitely many nodes in T extending σ is Σ_1.

So it is easy to show that any path through T can be computed from T'.

Harrington was inspired by the Jockush–Soare *low basis theorem*:

Theorem. There is a path P through T such that P' can be computed from T'.

The low basis theorem

The proof is an iterative construction. Define

$$\sigma_0 \subseteq \sigma_1 \subseteq \sigma_2 \subseteq \cdots$$

At stage e:

- Take another step through the tree.
- Try to restrict the tree so that for any path P, $\varphi_e^P(0) \uparrow$.

In the end, for any e, φ_e^P is defined at 0 if and only if this was *forced* at stage e.

Translated to the language of subsystems of second-order arithmetic, this says we can expand any model $\mathcal{M}$ by adding a path through a tree.

Conservativity of WKL_0 over RCA_0

To turn this into an interpretation of WKL_0 in RCA_0, we need to carry out the iteration internally.

- Hájek used a universal construction, but still needed a fiddly iteration.
- In my dissertation, I internalized an iterated forcing.
- In 2016, Tin Lok Wong used an arithmetized completeness theorem to avoid the iteration entirely.
- The book proof is a combination of Hájek's and Wong's approach.

Conservativity of WKL_0 over RCA_0

Let $A = \{(e, x) \mid \varphi_e^X(x) \downarrow = 0\}$ and $B = \{(e, x) \mid \varphi_e^X(x) \downarrow = 1\}$.

Theorem. A and B are universal computably inseparable c.e. sets in X, in the following sense: if A_0 and B_0 are any two sets that are disjoint and computably enumerable in X, then for any separation C of A from B there is a separation C_0 of A_0 from B_0 such that C_0 is many-one reducible to C.

Given the connection between computable inseparable c.e. sets and paths through trees, we have this following:

Theorem. For any set X, there is an infinite binary tree T computable from X, such that if T_0 is any infinite binary tree computable in X, a path P_0 through T_0 can be computed from any path P through T.

Conservativity of WKL_0 over RCA_0

This isn't enough: after we add P, we still have to worry about all the infinite binary trees computable from P.

Hájek wrote down a complicated formula describing the iterative construction.

In 2016, Wong used a path through an infinite binary tree to obtain a nonstandard model of arithmetic.

He showed that one can find a model of WKL_0 coded in that model.

Conservativity of WKL_0 over RCA_0

An ω-model of WKL_0 is a collection $\mathcal{S}$ of sets with the following property:

- Whenever X and Y are in $\mathcal{S}$, so is $X \oplus Y$.
- Whenever X is in $\mathcal{S}$ and $Y \leq_T X$, then Y is in $\mathcal{S}$.
- Whenever T is an infinite binary tree in $\mathcal{S}$, there is a P in $\mathcal{S}$ such that P is a path through T.

A set X of numbers can code a sequence of sets $(Y_i)_{i \in \mathbb{N}}$.

Conservativity of WKL_0 over RCA_0

Theorem. There is a computable infinite bintary tree T such that if P is any path through T, then the set $\mathcal{S} = \{(P)_i \mid i \in \mathbb{N}\}$ is an ω-model of WKL_0.

Proof. Just write down the requirements on P, in terms of finite approximations.

Relativizing the construction to an arbitrary set and formalizing it yields an interpretation of WKL_0 in RCA_0.