Formalizing forcing arguments in subsystems of second-order arithmetic

Ulrik Buchholtz

Stanford

April 26, 2011

Outline

Outline

Introduction

We will review the paper *Formalizing forcing arguments in subsystems of second-order arithmetic* by Jeremy Avigad.[2]

We will also draw on Avigad's survey article about forcing in proof theory.[1]

The mail result of [2] is an effective version of the following theorem:

Theorem (Harrington)

WKL_0 *is conservative over* RCA_0 *for* Π^1_1*-sentences.*

(Friedman had earlier shown conservativity for Π^0_2-sentences.)

The Brown-Simpson Extension

In fact, Avigad is able to treat the following extension:

Theorem (Brown-Simpson)

WKL+$_0$ *is conservative over* RCA$_0$ *for* Π^1_1*-sentences, where* WKL+$_0$ *is* WKL$_0$ *plus*

$$\forall n\, \forall \sigma\, \exists \tau \supset \sigma\, \varphi(n, \tau) \to \exists f\, \forall n\, \exists m\, \varphi(n, f[m]), \qquad \text{(BCT)}$$

where σ *and* τ *range over binary sequences,* f *is a function with range* $\{0, 1\}$ *and* $f[m]$ *denotes the sequence* $\langle f(0), \ldots, f(m-1)\rangle$*.*

(BCT) implies a version of the Baire Category Theorem, and the Open Mapping Theorem for separable Banach spaces.

Outline

Kripke structure

Recall that a Kripke structure for a first-order relational language consists of a tuple $\langle P, D, \Vdash \rangle$ where

- P is an inhabited poset, elements of which are callend "conditions,"
- D assigns to each $p \in P$ a set, $D(p)$, to be the "domain at p",
- for each relation symbol R and each $p \in P$, $p \Vdash R(\vec{a})$ denotes a relation on $D(p)$.

These data are required to satisfy the monotonicity requirements: for $q \leq p$ ("q is *stronger* than p"):

- $D(q) \supset D(p)$,
- if $p \Vdash R(\vec{a})$, then $q \Vdash R(\vec{a})$.

The classical forcing relation

1. $p \Vdash \varphi \wedge \psi$ if and only if $p \Vdash \varphi$ and $p \Vdash \psi$,
2. $p \Vdash \varphi \vee \psi$ if and only if $p \Vdash \varphi$ or $p \Vdash \psi$,
3. $p \Vdash \varphi \to \psi$ if and only if $\forall q \leq p\, (q \Vdash \varphi \to q \Vdash \psi)$,
4. $p \Vdash \forall x\, \varphi(x)$ if and only if $\forall q \leq p\, \forall a \in D(q)\, (q \Vdash \varphi(a))$,
5. $p \Vdash \exists x\, \varphi(x)$ if and only if $\exists a \in D(p)\, (p \Vdash \varphi(a))$.

Outline

Forcing syntactically

The above considerations take models at face value. By formalizing a forcing analysis of one theory T_1 inside another theory, T_2, we may be able to obtain stronger results, for instance concerning lengths of proofs.

To do this, define in T_2 predicates $\mathrm{Cond}(p)$, $q \leq p$, and $p \Vdash \mathrm{Name}(x)$ (of the displayed variables). Then we define, also in T_2, for each relation symbol R in the language of T_1, a relation $p \Vdash R(\vec{a})$. Finally, we prove in T_2 that this determines a Kripke structure where the axioms of T_1 are forced, and that forcing respects the logic of T_1.

Benefits of a syntactic forcing analysis

Assume the setup of the previous slide. Then the upshot is that whenever T_1 proves φ, T_2 proves that φ is forced.

Now, if T_2 proves that $\bot$ is not forced, then this shows that T_1 is consistent relative to T_2.

Further, if for a class $\mathcal{F}$ of formulae φ, T_2 proves that $\Vdash \varphi$ is equivalent to φ, then the interpretation shows that T_1 is conservative over T_2 relative to $\mathcal{F}$.

Examples of results

Let's consider a few examples of this approach, the first of which will be the focus of the presentation:

Recall that the theory WKL_0 extends RCA_0 with the following axiom

$$\forall T\,(T \text{ is an infinite binary tree} \rightarrow \exists P\,(P \text{ is a path through } T)).$$

It is an old result of Friedman that WKL_0 is conservative over RCA_0 for Π^0_2-sentences. Harrington strengthened this to Π^1_1-conservativity. These relied on model-theoretic arguments that gave no effective means of translating proofs using the above axiom to proofs without it.

Hájek [3] provided an effective version using recursion-theoretic coding techniques, whereas Avigad obtained an effective version by formalizing Harrington's forcing argument.

Goodman's theorem

Another example is provided by Beeson's version of Goodman's theorem:

Theorem

$\mathrm{HA}^\omega + (\mathrm{AC}) + (\mathrm{Ext})$ *is a conservative extension of* HA^ω.

Beeson formulated this as composition of:

- a realizability argument exploiting the fact that HA^ω proves that the axiom of choice is realizable, with
- the observation that in the negative fragment, "φ is realizable" is equivalent to φ, with
- a forcing argument adding "generic" functions to verify the axiom of choice, coding up witnesses to $\vee$- and $\exists$-subformulas.

Conservativity of $\mathsf{I}\Sigma_1$ over $\mathsf{I}\Sigma_1^{\mathsf{i}}$

As a final example, let us mention Thierry Coquand's observation that forcing can be used to show that $\mathsf{I}\Sigma_1$ is Π_2^0-conservative over its intuitionistic counterpart, $\mathsf{I}\Sigma_1^{\mathsf{i}}$.
Here, the double-negation translation isn't adequate by itself, since the translation of an instance of Σ_1-induction isn't again an instance of Σ_1-induction. However, adding Markov's principle

$$\neg\forall x\, A \to \exists x\, \neg A,$$

for quantifier-free A; translated Σ_1-sentences become equivalent to Σ_1-sentences. Thus, we need to interpret $\mathsf{I}\Sigma_1^{\mathsf{i}} + (\mathsf{MP})$ in $\mathsf{I}\Sigma_1^{\mathsf{i}}$.

Conservativity of $\mathsf{I}\Sigma_1$ over $\mathsf{I}\Sigma_1^{\mathsf{i}}$, continued

To do this, take conditions to be codes of finite sets of Π^0_1-sentences,

$$p = \ulcorner \{\forall x\, A_1(x), \forall x\, A_2(x), \dots, \forall x\, A_n(x)\} \urcorner,$$

with $p \leq q$ if and only if $p \supseteq q$. For atomic θ, define $p \Vdash \theta$ to be

$$\exists y\, (A_1(y) \wedge \dots \wedge A_n(y) \to \theta).$$

Then Markov's Principle is forced.

For further applications: see [1].

Outline

Forcing Weak Königs Lemma

In general outline, Harrington's argument starts with a model of RCA_0 and constructs a sequence of models

$$M = M_0 \subset_\omega M_1 \subset_\omega M_2 \subset_\omega \cdots \subset_\omega M_i \subset_\omega \cdots$$

where each M_i is a model of RCA_0, and if T is an infinite binary tree in M_i, there is a $j > i$ such that M_j contains an infinite path through T. Then $\cup M_i$ models WKL_0.

Avigad replicates this argument syntactically as follows.

Good strong forcing notions

Definition

Assume $\mathrm{Cond}(P)$ and $P \leq Q$ have been defined so that the base theory proves that the class of conditions forms a partial order. Definitions "$P \Vdash^{\mathsf{s}} \mathrm{Name}(x)$" and "$P \Vdash^{\mathsf{s}} \varphi$" (for atomic φ) form a *good strong forcing notion* if the following holds:

1. The free variables of $P \Vdash^{\mathsf{s}} \varphi$ are P together with those of φ; the free varables of $P \Vdash^{\mathsf{s}} \mathrm{Name}(X)$ are P and X,
2. Monotonicity; the base theory proves that $P \Vdash^{\mathsf{s}} \varphi$ and $P \Vdash^{\mathsf{s}} \mathrm{Name}(X)$ are monotone in P,
3. Substitution; for each term t,

$$P \Vdash^{\mathsf{s}} \varphi(x/t) \leftrightarrow (P \Vdash^{\mathsf{s}} \varphi)(x/t).$$

Good strong forcing notions extended

Definition

We extend a given good strong forcing notion to arbitrary formulas of $\mathcal{L}^2$ as follows:

1. $P \Vdash^{\mathsf{s}} \neg\varphi :\Leftrightarrow \forall Q \leq P \neg(Q \Vdash^{\mathsf{s}} \varphi)$,
2. $P \Vdash^{\mathsf{s}} \varphi \wedge \psi :\Leftrightarrow (P \Vdash^{\mathsf{s}} \varphi) \wedge (P \Vdash^{\mathsf{s}} \psi)$,
3. $P \Vdash^{\mathsf{s}} \varphi \vee \psi :\Leftrightarrow (P \Vdash^{\mathsf{s}} \varphi) \vee (P \Vdash^{\mathsf{s}} \psi)$,
4. $P \Vdash^{\mathsf{s}} \varphi \to \psi :\Leftrightarrow \forall Q \leq P\,(Q \Vdash^{\mathsf{s}} \varphi \to \exists R \leq Q\,(R \Vdash^{\mathsf{s}} \psi))$,
5. $P \Vdash^{\mathsf{s}} \exists x\,\varphi :\Leftrightarrow \exists x\,(P \Vdash^{\mathsf{s}} \varphi)$,
6. $P \Vdash^{\mathsf{s}} \forall x\,\varphi :\Leftrightarrow \forall x\,\forall Q \leq P\,\exists R \leq Q\,(R \Vdash^{\mathsf{s}} \varphi)$,
7. $P \Vdash^{\mathsf{s}} \exists X\,\varphi :\Leftrightarrow \exists X\,(P \Vdash^{\mathsf{s}} \mathrm{Name}(X) \wedge P \Vdash^{\mathsf{s}} \varphi)$,
8. $P \Vdash^{\mathsf{s}} \forall X\,\varphi :\Leftrightarrow \forall X\,\forall Q \leq P\,\exists R \leq Q\,(R \Vdash^{\mathsf{s}} \mathrm{Name}(X) \to R \Vdash^{\mathsf{s}} \varphi)$.

The extended forcing relation satisfies monotonicity and substitution.

Good weak forcing notions

A problem with good strong forcing notions is that they don't necessarily preserve logic! Basically this occurs when they're not "not-not stable," so we define:

Definition

A good strong forcing notion $\Vdash$ is a *good weak forcing notion* if the base theory additionally proves $P \Vdash \varphi \leftrightarrow P \Vdash \neg\neg\varphi$. The extension to arbitrary formulae differs from strong forcing in the following clauses:

3. $P \Vdash \varphi \vee \psi :\Leftrightarrow \forall Q \leq P \, \exists R \leq Q \, ((R \Vdash \varphi) \vee (R \Vdash \psi))$,
5. $P \Vdash \exists x \, \varphi :\Leftrightarrow \forall Q \leq P \, \exists R \leq Q \, \exists x \, (R \Vdash \varphi)$,
6. $P \Vdash \forall x \, \varphi :\Leftrightarrow \forall x \, (P \Vdash \varphi)$,
7. $P \Vdash \exists X \, \varphi :\Leftrightarrow \forall Q \leq P \, \exists R \leq Q \, \exists X \, (R \Vdash \mathrm{Name}(X) \wedge R \Vdash \varphi)$,
8. $P \Vdash \forall X \, \varphi :\Leftrightarrow \forall X \, (P \Vdash \mathrm{Name}(X) \rightarrow P \Vdash \varphi)$.

Good weak forcing notions, continued

An extended good weak forcing notion satisfies monotonicity, substitution and stability:

$$P \Vdash \varphi \leftrightarrow P \Vdash \neg\neg\varphi.$$

Lemma

Suppose $\Vdash^{\mathsf{s}}$ is a good strong forcing notion. Then the relation $\Vdash$ given by

$$P \Vdash \varphi :\Leftrightarrow P \Vdash^{\mathsf{s}} \neg\neg\varphi \qquad (*)$$

is a good weak forcing theory, and $(*)$ will hold as well for the extended relations.

$\frac{1}{2}$-forcing: one generic path

To force WKL, the first step is to add a generic path through some infinite tree. We do this in two steps: first we a single generic path ($\frac{1}{2}$-forcing); then we add all sets recursively definable from this new path and olds set, so as to model RCA_0 (1-forcing). Both will be *weak* forcing relations.

Definition

$1/2$-conditions are infinite binary trees:

$$\mathrm{Cond}_{\frac{1}{2}}(P) :\Leftrightarrow P \text{ is an binary tree} \wedge \forall n\, \exists \sigma \in P\, (\mathrm{len}(\sigma) = n),$$

(this is equivalent to a Π^0_1-formula), and we let $P \leq_{\frac{1}{2}} Q :\Leftrightarrow P \subset Q$.

The $\frac{1}{2}$-names are $\hat{X} = \{\langle 0, x\rangle \mid x \in X\}$ for old sets and $\hat{G} = \{\langle 1, 0\rangle\}$ for the new generic.

The $\frac{1}{2}$-forcing relation

We define:

$$P \Vdash_{\frac{1}{2}} t_1 = t_2 \quad :\Leftrightarrow \quad t_1 = t_2$$

$$P \Vdash_{\frac{1}{2}} t \in \hat{X} \quad :\Leftrightarrow \quad t \in X$$

$$P \Vdash_{\frac{1}{2}} t \in \hat{G} \quad :\Leftrightarrow \quad \exists n \, \forall \sigma \, (\sigma \in P \wedge \operatorname{len} \sigma = n \rightarrow t \subset \sigma).$$

The intuition of the last clause is that we want G to be an infinite path through P, so at some height n, all nodes have t as a prefix (thus, since P is prefix closed, all nodes of height greater than n will also have t as a prefix). Thus, all but finitely many nodes of P have t as a prefix.

This condition is Σ^0_1.

$\frac{1}{2}$-forcing: summing up

Now, the key facts about $\frac{1}{2}$-forcing are:

1. $\frac{1}{2}$-forcing is a good weak forcing notion,
2. RCA_0 proves that for φ not mentioning G, $\Vdash_{\frac{1}{2}} \varphi(\hat{X})$ is equivalent to $\varphi(X)$,
3. if φ is Σ^0_1 (resp. Π^0_2), then RCA_0 proves that $\Vdash_{\frac{1}{2}} \varphi$ is equivalent to another Σ^0_1 (resp. Π^0_2) formula.
4. RCA_0 proves that Σ^0_1-induction is $\frac{1}{2}$-generically valid.

The next step is to add names for all sets recursively definable from the $\hat{X}$ and $\hat{G}$.

1-forcing: recursive comprehension

We take 1-names to be triples $\langle X, \psi, \chi\rangle$, where $\psi(x, X, G)$ and $\chi(x, X, G)$ are codes of Σ^0_1 and Π^0_1-formulas determining a set which is recursive in X and G.

Let $\mathsf{Tr}_{\Sigma^0_1}$ and $\mathsf{Tr}_{\Pi^0_1}$ be suitable truth predicates. Then we define

$$P \Vdash_1 \mathrm{Name}(\langle X, \psi, \chi\rangle) :\Leftrightarrow P \Vdash_{\frac{1}{2}} \forall x \, (\mathsf{Tr}_{\Sigma^0_1}(\psi, x, \hat{X}, \hat{G}) \leftrightarrow \mathsf{Tr}_{\Pi^0_1}(\chi, x, \hat{X}, \hat{G})).$$

Then we can set

$$P \Vdash_1 t \in \langle X, \psi, \chi\rangle :\Leftrightarrow P \Vdash_{\frac{1}{2}} \mathsf{Tr}_{\Sigma^0_1}(\psi, t, \hat{X}, \hat{G}).$$

1-forcing: summing up

Now, the key facts about 1-forcing are:

1. 1-forcing is a good weak forcing notion,
2. RCA_0 proves that for φ not mentioning G, $\Vdash_1 \varphi(\hat{X})$ is equivalent to $\varphi(X)$,
3. if φ is Σ^0_1 (resp. Π^0_2), then RCA_0 proves that $\Vdash_1 \varphi$ is equivalent to another Σ^0_1 (resp. Π^0_2) formula.
4. RCA_0 proves that Σ^0_1-induction is 1-generically valid.
5. RCA_0 proves that each axiom of RCA_0 is 1-generically valid.
6. RCA_0 proves that if P is a 1-condition, then $P \Vdash_1 \exists X\,(X$ is an infinite path through $\hat{P})$.

From n- to $n+1$-forcing

Once n-forcing has been defined, we define:

1. An $n+1$-condition is a pair $\langle P, P' \rangle$ such that

$$\mathrm{Cond}_n(P) \wedge P \Vdash_n \mathrm{Name}(P') \wedge P \Vdash_n \mathrm{Cond}_1(P').$$

2. If $\langle P, P' \rangle$ and $\langle Q, Q' \rangle$ are $n+1$-conditions, then $\langle P, P' \rangle \leq_{n+1} \langle Q, Q' \rangle$ if and only if

$$P \leq_n P' \wedge P \Vdash_n (P' \leq_1 Q').$$

3. $\langle P, P' \rangle \Vdash_{n+1} \mathrm{Name}\, X$ if and only if

$$P \Vdash_n (P' \Vdash_1 \mathrm{Name}(X)).$$

4. $\langle P, P' \rangle \Vdash_{n+1} \varphi$ if and only if

$$P \Vdash_n (P' \Vdash_1 \varphi).$$

Uniform n-forcing

By carefully pushing the formula-complexity of n-forcing to Π^0_2, Avigad is able to find primitive recursive functions of n, giving the notions of condition, order, name, and "element of" as codes of Π^0_2-formulas (with n as a parameter).

Then, using the fact that RCA_0 is finitely axiomatizable, we can prove that "if $\bigwedge \mathsf{RCA}_0$ is n-forced, then $\bigwedge \mathsf{RCA}_0$ is $n+1$-forced."

Then we take an ω-condition to be an n-condition where $\bigwedge \mathsf{RCA}_0$ is n-forced.

ω-forcing: summing up

Now, the key facts about ω-forcing are:

1. ω-forcing is a good weak forcing notion,
2. RCA_0 proves that for φ not mentioning G, $\Vdash_\omega \varphi(\hat{X})$ is equivalent to $\varphi(X)$,
3. if φ is Σ^0_1 (resp. Π^0_2), then RCA_0 proves that $\Vdash_1 \varphi$ is equivalent to another Σ^0_1 (resp. Π^0_2) formula.
4. RCA_0 proves that Σ^0_1-induction is ω-generically valid.
5. RCA_0 proves that each axiom of RCA_0 is ω-generically valid.
6. RCA_0 proves that (WKL) is ω-generically valid.

The effective version of Brown-Simpson

More work forces also BCT, and by analyzing the transformation, we get

Theorem (Avigad)

There is a recursive function f and a polynomial p such that: if d codes a proof in WKL+$_0$ of a Π^1_1-formula φ, then $f(d)$ codes a proof of φ in RCA$_0$, and the length of $f(d)$ is less than p(length of d).

[1] Jeremy Avigad. "Forcing in proof theory". In: *Bull. Symbolic Logic* 10.3 (2004), pp. 305–333. ISSN: 1079-8986. DOI: `10.2178/bsl/1102022660`. URL: `http://dx.doi.org/10.2178/bsl/1102022660`.

[2] Jeremy Avigad. "Formalizing forcing arguments in subsystems of second-order arithmetic". In: *Ann. Pure Appl. Logic* 82.2 (1996), pp. 165–191. ISSN: 0168-0072. DOI: `10.1016/0168-0072(96)00003-6`. URL: `http://dx.doi.org/10.1016/0168-0072(96)00003-6`.

[3] Petr Hájek. "Interpretability and fragments of arithmetic". In: *Arithmetic, proof theory, and computational complexity (Prague, 1991)*. Vol. 23. Oxford Logic Guides. New York: Oxford Univ. Press, 1993, pp. 185–196.