

Forcing and the independence of the continuum hypothesis

Jesse Han

September 6, 2018

Abstract

In these notes, intended as the plaintext part of the Flypitch project, we give a complete account of the independence of the continuum hypothesis from ZFC, with special attention paid to comparing the different approaches: generic sets, Boolean-valued models, and double-negation sheaves.

1 Preliminaries

1.1 First-order logic

1.1.1 General logical symbols

Definition 1.1. We reserve the following general logical symbols:

| | |
|--|-------------|
| \neg | not |
| \vee | or |
| \wedge | and |
| \forall | for all |
| \exists | exists |
| $=$ | equals |
| $(,)$ | parentheses |
| $(x_i)_{i:\mathbb{N}}, (y_i)_{i:\mathbb{N}}, (z_i)_{i:\mathbb{N}}$ | variables |

1.1.2 First-order languages

Definition 1.2. A (first-order, one-sorted) **language** \mathcal{L} comprises the following data:

- (i) A collection of **constant symbols** $\text{Const}(\mathcal{L})$,
- (ii) a collection of **relation symbols** $\text{Rel}(\mathcal{L})$,
- (iii) a collection of **function symbols** $\text{Funct}(\mathcal{L})$, and
- (iv) an assignment of each symbol $S : \text{Const}(\mathcal{L}) \cup \text{Rel}(\mathcal{L}) \cup \text{Funct}(\mathcal{L})$ to a natural number $\text{arity}(S) : \mathbb{N}$.

Whenever we interpret a language on some carrier A , we mean for constants c to be interpreted as elements of $A^{\text{arity}(c)}$, relations R to be interpreted as subsets of $A^{\text{arity}(R)}$, and for function symbols to be interpreted as functions $A^{\text{arity}(f)} \rightarrow A$.

Example 1.3. • The language of groups comprises a 1-ary constant symbol for the identity and a 2-ary function for group multiplication.

- The language of rings comprises constant symbols 0 and 1 and 2-ary functions for addition and multiplication.
- The language of set theory comprises just one 2-ary relation \in .

1.1.3 Terms, formulas, and sentences

Definition 1.4. A **term** is a string of symbols defined by structural induction as follows:

1. Any variable v is a term.
2. Any constant c is a term.
3. If t_1, \dots, t_n are terms of arities a_1, \dots, a_n , then (t_1, \dots, t_n) is a term of arity $a_1 + \dots + a_n$.
4. If t is a term and f is a function symbol with matching arities, then ft is a term.

Whenever we interpret our language on a carrier A , we mean for terms to be interpreted as functions into A which we can construct by composing existing constants (constant functions), basic functions (i.e. the interpretations of the function symbols), and variables (identity).

Definition 1.5. A **formula** is defined by structural induction as follows:

1. If t_1 and t_2 are terms of the same arity, $t_1 = t_2$ is a formula.
2. If t is a term and R is a relation symbol, and t and R have the same arity, then Rt is a formula.
3. If φ is a formula, $\neg\varphi$ is a formula.
4. If φ and ψ are formulas, then $\varphi \vee \psi$ is a formula.
5. If φ and ψ are formulas, then $\varphi \wedge \psi$ is a formula.
6. If φ is a formula containing a variable v , then $\exists v\varphi$ is a formula.
7. If φ is a formula containing a variable v , then $\forall v\varphi$ is a formula.

Definition 1.6. Let φ be a formula containing the variables x_1, \dots, x_n . We say that the variable x_k is **free** if x_k is not contained in a subformula of the form $\exists x_k\psi$ or $\forall x_k\psi$.

x_k is **bound** if it is not free.

Definition 1.7. A formula is a **sentence** (or **statement**) if it contains no free variables.

We write $\text{Formulas}(\mathcal{L})$ for all the first-order formulas of \mathcal{L} , and we write $\text{Sentences}(\mathcal{L})$ for all the first-order sentences of \mathcal{L} .

By convention, we always include sentences called “true” and “false”.

1.1.4 Predicate calculus and provability

Definition 1.8. A **propositional function** is a function $f : \text{Prop}^k \rightarrow \text{Prop}$, for some $1 < k : \mathbb{N}$ which we define inductively as follows:

1. Each projection $(P_1, \dots, P_k) \mapsto P_j$ is a propositional function.
2. If f and g are propositional functions, so are

$$\neg f, f \wedge g, f \vee g, f \rightarrow g, \text{ and } f \leftrightarrow g,$$

where the operations above are carried out pointwise.

. f is a **tautology** if $\vdash \forall \vec{p} : \text{Prop}^k, f\vec{p} \leftrightarrow \text{true}$.

Every propositional function can be reified as a function $\text{Sentences}(\mathcal{L}) \rightarrow \text{Sentences}(\mathcal{L})$.

Definition 1.9. The **predicate calculus** comprises the following rules for deducing sentences from other sentences. We call deducible sentences **valid**.

- (a) (Rule of the propositional calculus) if \bar{f} is a reified tautology which takes k arguments, then for any k sentences A_1, \dots, A_k , the propositional combination $\bar{f}(\varphi_1, \dots, \varphi_k)$ is a valid sentence.
- (b) (Rule of modus ponens) If A and $A \rightarrow B$ are valid, then B is valid.
- (c) (Rules of equality)

(i) $\forall x, x = x, \forall x \forall y, x = y \wedge y = x$, and $\forall x \forall y \forall z, x = y \wedge y = z \rightarrow x = z$ are all valid.

(ii) Let $\varphi(x)$ be a formula whose only free variable is x . Then

$$\forall x \forall y, (x = y) \rightarrow (\varphi(x) \rightarrow \varphi(y))$$

is valid.

- (d) (Change of variable) If A is a sentence and A' represents A with all instances of a variable x switched to y , then $A \leftrightarrow A'$ is valid.
- (e) (Rule of specialization “ \forall -elimination”) Let c be any constant symbol, and let $\varphi(x)$ be a formula whose only free variable is x . Then $(\forall x \varphi(x)) \rightarrow \varphi(c)$ is valid.
- (f) $A \leftrightarrow (A \rightarrow \text{true})$ is valid.
- (g) (“ \neg -introduction”) If $\neg A \leftrightarrow (A \rightarrow \text{false})$ is valid.
- (h) (Generalization of constants “ \forall -introduction”) Let B be a sentence which does not contain the constant c or the variable x . Let $\varphi(x)$ be some formula such that $\varphi(x) \rightarrow B$ is valid. Then $\exists x \varphi(x) \rightarrow B$ is also valid.¹
- (i) (de Morgan laws) Let $\varphi(x)$ have x as its only free variable. Let B be a sentence which does not contain x . Then the following are valid statements:

$$(\neg(\forall x \varphi(x))) \leftrightarrow (\exists x \neg \varphi(x))$$

$$((\forall x \varphi(x)) \wedge B) \leftrightarrow ((\forall x (\varphi(x) \wedge B))$$

$$((\exists x \varphi(x)) \wedge B) \leftrightarrow ((\exists x (\varphi(x) \wedge B))$$

¹In particular, using the next rule, if $\neg \varphi(c) \rightarrow \text{false}$ is valid, so is $\exists x \neg \varphi(x) \rightarrow \text{false}$, so is $\neg \exists x \neg \varphi(x)$, and therefore so is $\forall x \varphi(x)$.

Definition 1.10. Let S be a collection of sentences.

- (i) We say that A is provable from S if there exist finitely many $B_1, \dots, B_n : S$ such that $(B_1 \wedge \dots \wedge B_n) \rightarrow A$ is valid.
- (ii) We say that S is consistent if false is not valid.

1.2 Models and satisfiability

For the remainder of this section we fix a language \mathcal{L} .

Definition 1.11. An \mathcal{L} -theory is a collection of sentences from $\text{Sentences}(\mathcal{L})$.

Definition 1.12. An \mathcal{L} -structure comprises the following data:

- (i) A carrier type A ,
- (ii) an assignment of every $c : \text{Const}(\mathcal{L})$ to a $c^A : A^{\text{arity}(c)}$,
- (iii) an assignment of every $R : \text{Rel}(\mathcal{L})$ to a subtype $R^A : A^{\text{arity}(R)} \rightarrow \text{Prop}$,
- (iv) an assignment of every $f : \text{Func}(\mathcal{L})$ to a function $f^A : A^{\text{arity}(f)} \rightarrow A$.

Definition 1.13. Let A be an \mathcal{L} -structure. Using the data of A being an \mathcal{L} -structure, we can inductively assign to every term t (of arity k and containing n free variables) a **realization** $\mathbf{r}(t) : A^n \rightarrow A^k$, as follows:

1. If $t = v$ for a variable v , $\mathbf{r}(t) = \text{id}_A = \lambda v. v$.
2. If $t = c$ for a constant symbol c , $\mathbf{r}(t) = A^0 \xrightarrow{c^A} A$.
3. If $t = (t_1, \dots, t_m)$, then $\mathbf{r}(t) = \mathbf{r}(t_1) \times \dots \times \mathbf{r}(t_m)$.
4. If $t = f(t_0)$ for some function symbol f , then $\mathbf{r}(t) = f^A \circ \mathbf{r}(t_0)$.

Definition 1.14. Let A be an \mathcal{L} -structure. Using the data of A being an \mathcal{L} -structure, we can inductively assign to every formula $\varphi(x_1, \dots, x_n)$ (where x_1, \dots, x_n exhaust the free variables of φ) a **realization** $\mathbf{r}(\varphi) : A^n \rightarrow \text{Prop}$, as follows:

1. If φ is of the form $t_1 = t_2$, then $\mathbf{r}(\varphi)$ is $\mathbf{r}(t_1) = \mathbf{r}(t_2)$ (where symbolic equality is realized as true equality).
2. If φ is of the form $R(t)$, $\mathbf{r}(R(t))$ is $R^A(\mathbf{r}(t))$.
3. If φ is of the form $\neg\psi$, then $\mathbf{r}(\varphi)$ is $\neg\mathbf{r}(\psi)$.
4. If φ is of the form $\psi \vee \psi'$, then $\mathbf{r}(\varphi)$ is $\mathbf{r}(\psi) \vee \mathbf{r}(\psi')$.
5. If φ is of the form $\psi \wedge \psi'$, then $\mathbf{r}(\varphi)$ is $\mathbf{r}(\psi) \wedge \mathbf{r}(\psi')$.
6. If φ is of the form $\exists v\psi$, then $\mathbf{r}(\varphi)$ is $\exists v\mathbf{r}(\psi)$.
7. If φ is of the form $\forall v\psi$, then $\mathbf{r}(\varphi)$ is $\forall v\mathbf{r}(\psi)$.

In particular, each sentence φ is sent to a $\text{Prop } \mathbf{r}(\varphi)$.

Definition 1.15. Let A be an \mathcal{L} -structure, and let φ be a sentence. We say that A **satisfies** φ , written

$$A \models \varphi,$$

if $\vdash \mathbf{r}(\varphi)$.

Definition 1.16. Let T be an \mathcal{L} -theory, and let A be an \mathcal{L} -structure. We say that A is a **model** of T if for every sentence $\varphi : T$, $A \models \varphi$.

1.3 The completeness theorem

TODO(jesse)

1.4 The Löwenheim-Skolem theorem

TODO(kody)

2 Generic sets: Cohen's original proof

3 Boolean-valued models

4 Sheaves and filterquotients