

→ ACの相対的無矛盾性証明 の Isabelle/ZF による形式化

東北大学 大学院情報科学研究科

住井・松田研究室 M2

舟根大喜

November 22, 2024

概要

やったこと

¬AC の ZF 上の相対的無矛盾性証明を
Isabelle/ZF で形式化

動機

- 公理的集合論の重要な技法である強制法 (後述) を用いた議論の形式化は限られている
- AC の相対的無矛盾性証明は形式化されており合わせて AC の独立性が形式化されたことになる

用語

- **公理的集合論** :
集合がみたすべき性質を公理として定めた集合論
- **ZF** : Zermelo-Fraenkel 公理系。一階述語論理上で形式化されたよく採用される公理系
- **AC** : 選択公理 (Axiom of Choice)
任意の非空集合の族からそれぞれ1つの元を選ぶ関数が存在するという公理
 - 整列可能定理、Zorn の補題など同値な重要な命題がある

用語

- *Extensionality*: $\forall x \forall y (\forall z (z \in x \leftrightarrow z \in y) \rightarrow x = y)$
- *Pairing*: $\forall x \forall y \exists z \forall w (w \in z \leftrightarrow w = x \vee w = y)$
- *Union*: $\forall x \exists y \forall z (z \in y \leftrightarrow \exists w (z \in w \wedge w \in x))$
- *Power set*: $\forall x \exists y \forall z (z \in y \leftrightarrow z \subseteq x)$
- *Infinity*: $\exists x (\emptyset \in x \wedge \forall y (y \in x \rightarrow y \cup \{y\} \in x))$
- *Regularity*: $\forall x (x \neq \emptyset \rightarrow \exists y (y \in x \wedge y \cap x = \emptyset))$
- *Infinity*: $\forall x (x \neq \emptyset \rightarrow \exists y (y \in x \wedge \forall z (z \in x \rightarrow z \notin y)))$
- *Separation*: $\forall p \forall x \exists y \forall z (z \in y \leftrightarrow z \in x \wedge \phi(z, p))$
- *Replacement*: $\forall p (\forall x \forall y \forall z (\phi(x, y, p) \wedge \phi(x, z, p) \rightarrow y = z) \rightarrow \forall X \exists Y \forall y (y \in Y \leftrightarrow \exists x (x \in X \wedge \phi(x, y, p))))$
- *Choice*: $\forall x \exists f ("f \text{ is a function on } x" \wedge \forall y (y \in x \rightarrow f(y) \in y))$

- 命題 φ が公理系 T 上で**相対的無矛盾**：
 T が無矛盾ならば公理系 $T + \varphi$ も無矛盾なこと
- φ が T から**独立**：
 T から φ も $\neg\varphi$ も証明できないこと
- ▶ φ と $\neg\varphi$ の T 上の相対的無矛盾性を示すことで
 φ が T から独立であることが証明できる
(T が無矛盾と仮定すれば)

用語

■ 強制法 :

集合論のモデルを拡張し新しいモデルを作る技法

- M を強制法で拡張したモデルを
 M の **generic extension** と呼び $M[G]$ と書く
- $M[G]$ は poset \mathbb{P} と generic filter G に依存しており、 \mathbb{P} をうまく選ぶことで、ある程度 $M[G]$ で成り立つことを制御できる
- \mathbb{P} を決めたとえで、強制関係 \Vdash によって、 $M[G]$ で成り立つことを確認できる

※モデルが存在すれば無矛盾であるため、強制法でモデルを構成することで無矛盾性を証明できる

Isabelle/ZF

Isabelle/ZF における先行研究

- CH の ZFC 上の独立性証明 [Gunther et al. 20,22]
 - 強制法の形式化 (13K 行)
 - CH の独立性証明 (16K 行) ※ Lean3 にもある
- AC の ZF 上の相対的無矛盾性証明 [Paulson 02]
 - 構成可能宇宙を形式化 (12K 行)
- ▶ \neg AC の相対的無矛盾性証明は
形式化されていなかったもので挑戦

本研究における Isabelle/ZF の利点

- 集合論に関する補題・糖衣構文が豊富
 - 強制法の形式化 [Gunther et al. 20] が
すでにある
 - ZF の c.t.m. の存在を仮定している
 - c.t.m. : countable transitive model
- ※ Lean3 にも強制法の形式化がある

本研究では Isabelle/ZF 上で
さらに形式化された ZF を扱う

- 今後出てくる「ZF のモデル」とは、Isabelle/ZF 上で形式化された ZF を満たす集合のこと
- [Paulson 02] の形式化を利用

[Paulson 02] の形式化

text <De Bruijn representation.

Unbound variables get their denotations from an environment.>

consts formula :: i

datatype

```
"formula" = Member ("x ∈ nat", "y ∈ nat")
            | Equal ("x ∈ nat", "y ∈ nat")
            | Nand ("p ∈ formula", "q ∈ formula")
            | Forall ("p ∈ formula")
```

consts satisfies :: "[i,i]⇒i"

primrec (*explicit lambda is required because the environment varies*)

```
"satisfies(A, Member(x,y)) =
  (λenv ∈ list(A). bool_of_o (nth(x,env) ∈ nth(y,env)))"
```

```
"satisfies(A, Equal(x,y)) =
  (λenv ∈ list(A). bool_of_o (nth(x,env) = nth(y,env)))"
```

```
"satisfies(A, Nand(p,q)) =
  (λenv ∈ list(A). not ((satisfies(A,p)`env) and (satisfies(A,q)`env)))"
```

```
"satisfies(A, Forall(p)) =
  (λenv ∈ list(A). bool_of_o (∀x∈A. satisfies(A,p) ` (Cons(x,env)) = 1))"
```

証明概略

証明概略 [Karagila 23]

ZF の c.t.m. M から出発して
ZF + \neg AC のモデル N を構成する

- N は First Cohen Model と呼ばれるモデル
 - generic extension の部分モデルである
symmetric extension のひとつ
- N は ZF を満たすが、整列可能定理を満たさない
 - $N \models$ 「単射 $\omega \rightarrow A$ がない無限集合 A が存在」
 - N ではこの A が整列できない

Isabelle/ZF による形式化

本研究で形式証明した命題

```
theorem ZF_notAC_main_theorem :  
  fixes M  
  assumes "nat  $\approx$  M" "M  $\models$  ZF" "Transset(M)"  
  shows " $\exists N. \text{nat} \approx N \wedge N \models \text{ZF} \wedge \text{Transset}(N)$   
     $\wedge \neg(\forall A \in N. \exists r \in N. \text{wellordered}(\#N, A, r))$ "
```

意味

M を ZF の c.t.m. とする。このとき
ある ZF の c.t.m. N があって
 N は整列可能定理を満たさない

以下の工程に分けられる

1. symmetric extension の構成法の定義
2. ZF のモデルであることの証明
3. First Cohen Model の構成
4. それが $\neg AC$ を満たすことの証明

作業量

約1万5千行のコード 補題など (3K 行)

1. symmetric extension の定義 (3K 行)
2. ZF のモデルであることの証明 (5K 行)
3. First Cohen Model の構成 (2K 行)
4. それが $\neg AC$ を満たすことの証明 (2K 行)

※ Isabelle/ZF での集合論の形式化の各先行研究と
同じくらい

面倒だった点 自明なことの確認が大変

- クラスが本当にクラスであること
 - 実際に論理式を構成する必要がある
- 定義した関数が本当に関数であること
 - 特に「帰納的に定義された M 内の関数」
 - 仮定と ZF からちゃんと構成できるか？
 - このような関数を定義するための補題が 2 千行以上

困難だった点 N が ZF をみたすことの証明

Isabelle/ZF で構成した N が $M[G]$ においてクラスであることが証明できなかった

- 書き下すのが面倒だけでなく、非自明？
- これは [Karagila 23] で用いられている次の命題の証明に必要

命題

N が推移的かつ almost universal なクラスで Δ_0 -separation を満たすならば N は ZF の内部モデルである

解決策

- HS に相対化した強制関係 \Vdash_{HS} を形式化
 - 参考資料 [Karagila 23] に書かれている概念
 - 強制関係の定義の量化の動く範囲を HS に制限
 - \Vdash_{HS} は、symmetric extension に対し、generic extension に対する \Vdash のように振舞う
- \Vdash_{HS} を用いて ZF のモデルであることを証明

```
lemma HS_truth_lemma:
```

```
  assumes
```

```
    "φ ∈ formula" "M_generic(G)"
```

```
  shows
```

```
    "∧ env. env ∈ list(HS) ⇒ arity(φ) ≤ length(env) ⇒  
     (∃ p ∈ G. p ⊩_HS φ env) ⇔ SymExt(G, map(val(G), env) ⊨ φ"
```

```
lemma definition_of_forcing_HS:
```

```
  assumes
```

```
    "p ∈ P" "φ ∈ formula" "env ∈ list(HS)" "arity(φ) ≤ length(env)"
```

```
  shows
```

```
    "(p ⊩_HS φ env) ⇔  
     (∀ G. M_generic(G) ∧ p ∈ G ⇒ SymExt(G, map(val(G), env) ⊨ φ)"
```

考察

考察(1) c.t.m. アプローチについて

- 本研究で形式化したのは、
「ZF の c.t.m. が存在 \rightarrow ZF+ \neg AC の c.t.m. が存在」
 - ZF の c.t.m. の存在は、強制法の形式化
[Gunther et al. 20] を使うため仮定
- 証明したいのは $\text{Con}(\text{ZF}) \rightarrow \text{Con}(\text{ZF} + \neg \text{AC})$ だが、
ZF の c.t.m. の存在は、 $\text{Con}(\text{ZF})$ から証明できない
 - このギャップを埋めることができるが、
通常 c.t.m. アプローチの紙の上での証明では大まかな議論のみで省略される
 - この部分の形式化できていない
(本当は形式化したい)

形式化できていない部分

以下の形式化ができれば、
ZF の c.t.m. の存在の仮定をなくせる

- 任意の ZF の有限部分 Δ に対し、ZF の有限部分 Γ があって Γ の c.t.m. が存在すれば $\Delta + \neg AC$ のモデルが存在する
 - 今回の形式化を修正すれば可能
- ZF の有限部分 Γ に対し、 Γ の c.t.m. が存在する
 - ZF モデルの中で ZF のモデルを考える必要がある？
 - 形式化が難しい？

考察(2) メタ/対象レベル

ZF のモデルの中の性質の証明が大変だった

- コード化された論理式を扱う必要があった
- メタレベルで成り立つことをもう一度証明しなければいけなかった
- ▶ メタ/対象レベルの証明を同時に書ける or 他方に変換できるような機能があると便利
- ▶ 今回のテーマに限らず
数学基礎論の形式化でも有用

まとめ

\neg AC の相対無矛盾性証明を Isabelle/ZF で形式化

- ZF の c.t.m. から出発し、
ZF + \neg AC をみたす symmetric extension を構成
- 紙の上では省略される c.t.m. に関する議論の形式化が残っている
- 参考資料の通りにいかず試行錯誤した部分も
- メタ/対象レベルの形式的証明を「つなげる」機能がほしい

参考文献 (1)

- K. Kunen, Set Theory An Introduction To Independence Proofs, North-Holland, 1980
日本語訳: 藤田 博司 訳, 集合論: 独立性証明への案内, 日本評論社, 2008
- T. Jech, Set Theory: The Third Millennium Edition, Springer, 2002
- T. Jech, The Axiom of Choice, Dover Publications, 2008
- A. Karagila, Lecture Notes: Forcing & Symmetric Extensions, 2023

参考文献 (2)

- G. Klein et al., seL4: Formal Verification of an OS Kernel, 2014
- LC. Paulson, The Relative Consistency of the Axiom of Choice Mechanized Using Isabelle/ZF, 2003
- E. Gunther et al., Formalization of Forcing in Isabelle/ZF, 2020
- E. Gunther et al., The Independence of the Continuum Hypothesis in Isabelle/ZF, 2022