La independencia de CH formalizada en Isabelle/ZF

P. Sánchez Terraf[1]

Trabajo conjunto con Emmanuel Gunther, Miguel Pagano y Matías Steinberg

CIEM-FaMAF — Universidad Nacional de Córdoba

Reunión Anual UMA

Neuquén, 21 / 09 / 2022

[1] Financiado por proyecto Secyt-UNC 33620180100465CB y Conicet.

Verificación

La **verificación formal**: uso de *asistentes de prueba* para chequear que una demostración es correcta al máximo nivel de detalle.
No se omite ningún detalle, reduciendo el resultado a probar hasta los mismos axiomas.

Verificación

La **verificación formal**: uso de *asistentes de prueba* para chequear que una demostración es correcta al máximo nivel de detalle.
No se omite ningún detalle, reduciendo el resultado a probar hasta los mismos axiomas.

Atención

$$\text{Prueba formalizada} \neq \text{Prueba automática}$$

Verificación

La **verificación formal**: uso de *asistentes de prueba* para chequear que una demostración es correcta al máximo nivel de detalle.
No se omite ningún detalle, reduciendo el resultado a probar hasta los mismos axiomas.

Atención

$$\text{Prueba formalizada} \neq \text{Prueba automática}$$

Herramientas

Agda, Coq, HOL Light, Lean, ACL2, Isabelle.
Tienen dispobibles varias lógicas [Avigad, 2020].

Verificación

Proyectos notables de formalización

- Teorema de los Cuatro Colores [Gonthier, 2008], Teorema de Orden Impar [Gonthier et al., 2013].
- Matemática Univalente / Teoría de tipos homotópica [Univalent Foundations Program, 2013], concebido por Voevodsky.
- El proyecto *Flyspeck* [Hales et al., 2017, Forum Math. Pi] sobre la Conjetura de Kepler.

Verificación

Proyectos notables de formalización

- Teorema de los Cuatro Colores [Gonthier, 2008], Teorema de Orden Impar [Gonthier et al., 2013].
- Matemática Univalente / Teoría de tipos homotópica [Univalent Foundations Program, 2013], concebido por Voevodsky.
- El proyecto *Flyspeck* [Hales et al., 2017, Forum Math. Pi] sobre la Conjetura de Kepler.

Grandes proyectos presentes

- Isabelle: HOL, AFP. Ver también Paulson [2018].
- Lean: proyectos sobre `mathlib`, promovidos por K. Buzzard [mathlib Community, 2020].

El enfoque ctm al forcing, 1/4

Modelo contable transitivo (ctm) de ZF

$\langle M, E\rangle \models ZF$ donde

- M es *estándar*: $E := \in \restriction M$.
- M es contable y *transitivo*: $x \in y \in M \implies x \in M$.

El enfoque ctm al forcing, 1/4

Modelo contable transitivo (ctm) de ZF

$\langle M, E\rangle \models ZF$ donde

- M es *estándar*: $E := \in \restriction M$.
- M es contable y *transitivo*: $x \in y \in M \implies x \in M$.

Notar. Si $\langle N, R\rangle \models ZF$ con R bien fundada, entonces existe un ctm M de ZF.

El enfoque ctm al forcing, 1/4

Modelo contable transitivo (ctm) de ZF

$\langle M, E\rangle \models ZF$ donde

- M es *estándar*: $E := \in \restriction M$.
- M es contable y *transitivo*: $x \in y \in M \implies x \in M$.

Notar. Si $\langle N, R\rangle \models ZF$ con R bien fundada, entonces existe un ctm M de ZF.
Tiene sentido comparar, para $x, y \in M$:

$$x \subseteq y \qquad \text{y} \qquad M \models x \subseteq y$$

El enfoque ctm al forcing, 1/4

Modelo contable transitivo (ctm) de ZF

$\langle M, E\rangle \models ZF$ donde

- M es *estándar*: $E := \in \restriction M$.
- M es contable y *transitivo*: $x \in y \in M \implies x \in M$.

Notar. Si $\langle N, R\rangle \models ZF$ con R bien fundada, entonces existe un ctm M de ZF.
Tiene sentido comparar, para $x, y \in M$:

$$x \subseteq y \qquad \text{y} \qquad M \models x \subseteq y$$

El lado derecho se puede escribir como:

$$\forall z.\ z \in M \longrightarrow (z \in x \longrightarrow z \in y), \qquad \text{la } \textbf{relativización } \subseteq^M \text{ de } \subseteq \text{ a } M.$$

El enfoque ctm al forcing, 1/4

Modelo contable transitivo (ctm) de ZF

$\langle M, E\rangle \models ZF$ donde

- M es *estándar*: $E := \in \restriction M$.
- M es contable y *transitivo*: $x \in y \in M \implies x \in M$.

Notar. Si $\langle N, R\rangle \models ZF$ con R bien fundada, entonces existe un ctm M de ZF.
Tiene sentido comparar, para $x, y \in M$:

$$x \subseteq y \iff M \models x \subseteq y$$

El lado derecho se puede escribir como:

$\forall z.\ z \in M \longrightarrow (z \in x \longrightarrow z \in y)$, la **relativización** $\subseteq^M$ de $\subseteq$ a M.

En este caso, sabemos que $\subseteq$ es **absoluta para modelos transitivos**.

El enfoque ctm al forcing, 2/4

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una *noción de forzamiento* (un cuasiorden con máximo). Dado un filtro M*-genérico* $G \subseteq \mathbb{P}$, podemos agregrarlo a M para formar la **extensión genérica** $M[G]$.

El enfoque ctm al forcing, 2/4

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una *noción de forzamiento* (un cuasiorden con máximo).
Dado un filtro M*-genérico* $G \subseteq \mathbb{P}$, podemos agregrarlo a M para formar la **extensión genérica** $M[G]$.
Cada $a \in M[G]$ está codificado por algún $\dot{a} \in M$ a través de la función val:

$$M[G] := \{val(G, \dot{a}) : \dot{a} \in M\}$$

El enfoque ctm al forcing, 2/4

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una *noción de forzamiento* (un cuasiorden con máximo).
Dado un filtro M*-genérico* $G \subseteq \mathbb{P}$, podemos agregrarlo a M para formar la **extensión genérica** $M[G]$.
Cada $a \in M[G]$ está codificado por algún $\dot{a} \in M$ a través de la función val:

$$M[G] := \{val(G, \dot{a}) : \dot{a} \in M\}$$

Fundamentalmente, la **validez** en $M[G]$ está codificada en M por cierta función `forces`.

El enfoque ctm al forcing, 3/4

Beneficios de usar ctms

1. La contabilidad asegura la existencia de genéricos (por el Lema de Rasiowa-Sikorski).

El enfoque ctm al forcing, 3/4

Beneficios de usar ctms

1. La contabilidad asegura la existencia de genéricos (por el Lema de Rasiowa-Sikorski).
2. La absolutez proveee un tratamiento transparente de muchos conceptos.

$$\alpha \text{ es ordinal} \iff M \models \text{“}\alpha \text{ es ordinal”} \iff M[G] \models \text{“}\alpha \text{ es ordinal”}$$

El enfoque ctm al forcing, 3/4

Beneficios de usar ctms

1. La contabilidad asegura la existencia de genéricos (por el Lema de Rasiowa-Sikorski).
2. La absolutez proveee un tratamiento transparente de muchos conceptos.

   α es ordinal $\iff M \models$ "α es ordinal" $\iff M[G] \models$ "α es ordinal"
3. Tanto M y $M[G]$ son modelos tradicionales (2-valorados).

El enfoque ctm al forcing, 4/4

Eligiendo $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle$ apropiadamente uno puede ajustar finamente las propiedades de primer orden de $M[G]$ (para cada genérico G).

El enfoque ctm al forcing, 4/4

Eligiendo $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle$ apropiadamente uno puede ajustar finamente las propiedades de primer orden de $M[G]$ (para cada genérico G).

Theorem ([Cohen, 1963])

Si $\mathbb{P}$ es el conjunto de funciones parciales binarias finitas con dominio incluido en $\aleph_2^M$, $M[G]$ satisface la negación de la ***Hipótesis del Continuo*** *(CH):*

$$M[G] \models 2^{\aleph_0} > \aleph_1.$$

ZF-Constructible

La “lógica” Isabelle/ZF llega hasta el Teorema de Hessenberg $|A| \times |A| = |A|$.
Nuestra decisión de usar Isabelle (durante 2017) fue inducida por su *librería* de constructibilidad [Paulson, 2003].

ZF-Constructible

La “lógica” Isabelle/ZF llega hasta el Teorema de Hessenberg $|A| \times |A| = |A|$.
Nuestra decisión de usar Isabelle (durante 2017) fue inducida por su *librería* de constructibilidad [Paulson, 2003].
Contiene:

- un desarrollo de la relativización y absolutez para clases C;
- la construcción del conjunto de `formula` y la relación de satisfacción;
- una versión de Principio de Reflexión (Levy-Montague); y
- El desarrollo de L y la prueba de que satisface AC.

ZF-Constructible

La “lógica” Isabelle/ZF llega hasta el Teorema de Hessenberg $|A| \times |A| = |A|$.
Nuestra decisión de usar Isabelle (durante 2017) fue inducida por su *librería* de constructibilidad [Paulson, 2003].
Contiene:

- un desarrollo de la relativización y absolutez para clases C;
- la construcción del conjunto de `formula` y la relación de satisfacción;
- una versión de Principio de Reflexión (Levy-Montague); y
- El desarrollo de L y la prueba de que satisface AC.

Disciplina de relativización y síntesis

I $p = \mathtt{Pow}(x)$ :: `i` (término original).

ZF-Constructible

La “lógica” Isabelle/ZF llega hasta el Teorema de Hessenberg $|A| \times |A| = |A|$.
Nuestra decisión de usar Isabelle (durante 2017) fue inducida por su *librería* de constructibilidad [Paulson, 2003].
Contiene:

- un desarrollo de la relativización y absolutez para clases C;
- la construcción del conjunto de `formula` y la relación de satisfacción;
- una versión de Principio de Reflexión (Levy-Montague); y
- El desarrollo de L y la prueba de que satisface AC.

Disciplina de relativización y síntesis

I $p = \mathtt{Pow}(x) :: \mathtt{i}$ (término original).

II $\mathtt{is\_Pow}(C, x, p) :: \mathtt{o}$ (completamente relacional).

ZF-Constructible

La “lógica” Isabelle/ZF llega hasta el Teorema de Hessenberg $|A| \times |A| = |A|$.
Nuestra decisión de usar Isabelle (durante 2017) fue inducida por su *librería* de constructibilidad [Paulson, 2003].
Contiene:

- un desarrollo de la relativización y absolutez para clases C;
- la construcción del conjunto de `formula` y la relación de satisfacción;
- una versión de Principio de Reflexión (Levy-Montague); y
- El desarrollo de L y la prueba de que satisface AC.

Disciplina de relativización y síntesis

I $p = \mathtt{Pow}(x) :: \mathtt{i}$ (término original).

II $\mathtt{is\_Pow}(C, x, p) :: \mathtt{o}$ (completamente relacional).

III $\mathtt{is\_Pow\_fm}(0, 1, 2) :: \mathtt{i}$ (elemento de `formula`).

Índices de De Bruijn

Proveen un modo de escribir fórmulas de primer orden que simplifica el manejo de las variables ligadas.

$$(\forall\,.\,0 \in 1 \longrightarrow 0 \in 2)$$

Índices de De Bruijn

Proveen un modo de escribir fórmulas de primer orden que simplifica el manejo de las variables ligadas.

$$(\forall\,.\,0 \in 1 \longrightarrow 0 \in 2)$$

- Las asignaciones vienen dadas por listas de elementos del modelo.

$$A, [x, y, z, w] \models 0 \in 1 \longrightarrow 0 \in 2 \iff (x \in y \text{ implica } x \in z)$$

Índices de De Bruijn

Proveen un modo de escribir fórmulas de primer orden que simplifica el manejo de las variables ligadas.

$$(\forall\,.\,0 \in 1 \longrightarrow 0 \in 2)$$

- Las asignaciones vienen dadas por listas de elementos del modelo.

$$A, [x, y, z, w] \models 0 \in 1 \longrightarrow 0 \in 2 \iff (x \in y \text{ implica } x \in z)$$

- Los cuantificadores desplazan el indizado. El índice nos dice “cuántos cuantificadores saltar”.

$$A, [y, z, w] \models (\forall\,.\,0 \in 1 \longrightarrow 0 \in 2) \iff (y \subseteq z)$$

Los Teoremas Fundamentales

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una noción de forzamiento. Dado $G \subseteq \mathbb{P}$, tenemos
$M[G] := \{val(P, G, \dot{a}) : \dot{a} \in M\}$
Ésta es la versión de los teoremas de forcing en nuestra formalización.

Los Teoremas Fundamentales

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una noción de forzamiento. Dado $G \subseteq \mathbb{P}$, tenemos $M[G] := \{val(P, G, \dot{a}) : \dot{a} \in M\}$
Ésta es la versión de los teoremas de forcing en nuestra formalización.

Theorem (Cohen [1963])

Hay una función $\texttt{forces} : \texttt{formula} \to \texttt{formula}$ *tal que para cada* φ *y* $\tau_0, \ldots, \tau_n \in M$,

Lema de la Verdad *para cada* G M*-genérico,*

$$M[G], [val(P, G, \tau_0), \ldots, val(P, G, \tau_n)] \models \varphi$$
$$\iff$$
$$\exists p \in G.\ \ M, [p, \mathbb{P}, \preceq, \mathbb{1}, \tau_0, \ldots, \tau_n] \models \texttt{forces}(\varphi).$$

Los Teoremas Fundamentales

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una noción de forzamiento. Dado $G \subseteq \mathbb{P}$, tenemos $M[G] := \{val(P, G, \dot{a}) : \dot{a} \in M\}$
Ésta es la versión de los teoremas de forcing en nuestra formalización.

Theorem (Cohen [1963])

Hay una función $\texttt{forces} : \texttt{formula} \to \texttt{formula}$ *tal que para cada* φ *y* $\tau_0, \ldots, \tau_n \in M$,

Lema de la Verdad *para cada* G M*-genérico,*

$$M[G], [val(P, G, \tau_0), \ldots, val(P, G, \tau_n)] \models \varphi$$
$$\iff$$
$$\exists p \in G.\ \ M, [p, \mathbb{P}, \preceq, \mathbb{1}, \tau_0, \ldots, \tau_n] \models \texttt{forces}(\varphi).$$
$$\searrow \quad p \Vdash \varphi\ [\tau_0, \ldots, \tau_n] \quad \swarrow$$

Los Teoremas Fundamentales

Sea $\langle \mathbb{P}, \preceq, \mathbb{1} \rangle \in M$ una noción de forzamiento. Dado $G \subseteq \mathbb{P}$, tenemos $M[G] := \{val(P, G, \dot{a}) : \dot{a} \in M\}$
Ésta es la versión de los teoremas de forcing en nuestra formalización.

Theorem (Cohen [1963])

Hay una función $\texttt{forces} : \texttt{formula} \to \texttt{formula}$ *tal que para cada* φ *y* $\tau_0, \ldots, \tau_n \in M$,

Lema de la Verdad *para cada* G M*-genérico,*

$$M[G], [val(P, G, \tau_0), \ldots, val(P, G, \tau_n)] \models \varphi$$
$$\iff$$
$$\exists p \in G.\ M, [p, \mathbb{P}, \preceq, \mathbb{1}, \tau_0, \ldots, \tau_n] \models \texttt{forces}(\varphi).$$
$$\searrow \quad p \Vdash \varphi\ [\tau_0, \ldots, \tau_n] \quad \swarrow$$

Lema de Densidad $p \Vdash \varphi\ [\tau_0, \ldots, \tau_n] \iff \{q \in \mathbb{P} : q \Vdash \varphi\ [\tau_0, \ldots, \tau_n]\}$ *es denso bajo* p.

Logros

¿Qué formalizamos?

- Un tratamiendo de relativización *in extenso*.

Logros

¿Qué formalizamos?

- Un tratamiendo de relativización *in extenso*.
  - Resultados faltantes (desde cardinales sucesores, conjuntos contables)
  - Relativizamos los existentes.
  - Optimizamos ZF-Constructible (ahora parte de Isabelle).

Logros

¿Qué formalizamos?

- Un tratamiendo de relativización *in extenso*.
  - Resultados faltantes (desde cardinales sucesores, conjuntos contables)
  - Relativizamos los existentes.
  - Optimizamos ZF-Constructible (ahora parte de Isabelle).
- Una presentación modelo-teórica del forcing.
  - Extensiones propias de ctms de ZF.
  - Construcción de ctms de $ZFC + \neg CH$ y de $ZFC + CH$ dado uno ZFC.

```
theorem ctm_of_not_CH:
  assumes
    "M ≈ ω" "Transset(M)" "M ⊨ ZFC"
  shows
    "∃N.
      M ⊆ N ∧ N ≈ ω ∧ Transset(N) ∧ N ⊨ ZFC ∪ {·¬·CH··} ∧
      (∀α. Ord(α) ⟶ (α ∈ M ⟷ α ∈ N))"
```

Logros

¿Qué formalizamos?

- Un tratamiendo de relativización *in extenso*.
  - Resultados faltantes (desde cardinales sucesores, conjuntos contables)
  - Relativizamos los existentes.
  - Optimizamos ZF-Constructible (ahora parte de Isabelle).
- Una presentación modelo-teórica del forcing.
  - Extensiones propias de ctms de ZF.
  - Construcción de ctms de $ZFC + \neg CH$ y de $ZFC + CH$ dado uno ZFC.

No incluimos nociones metateóricas (cálculo de primer orden, etc).

Logros

¿Qué formalizamos?

- Un tratamiendo de relativización *in extenso*.
  - Resultados faltantes (desde cardinales sucesores, conjuntos contables)
  - Relativizamos los existentes.
  - Optimizamos ZF-Constructible (ahora parte de Isabelle).
- Una presentación modelo-teórica del forcing.
  - Extensiones propias de ctms de ZF.
  - Construcción de ctms de $ZFC + \neg CH$ y de $ZFC + CH$ dado uno ZFC.

No incluimos nociones metateóricas (cálculo de primer orden, etc).

¿Qué se puede deducir de la presente formalización?

- Cuáles instancias de Reemplazo se necesitan para poner a andar la maquinaria del forcing.

Logros

¿Qué formalizamos?

- Un tratamiendo de relativización *in extenso*.
  - Resultados faltantes (desde cardinales sucesores, conjuntos contables)
  - Relativizamos los existentes.
  - Optimizamos ZF-Constructible (ahora parte de Isabelle).
- Una presentación modelo-teórica del forcing.
  - Extensiones propias de ctms de ZF.
  - Construcción de ctms de $ZFC + \neg CH$ y de $ZFC + CH$ dado uno ZFC.

No incluimos nociones metateóricas (cálculo de primer orden, etc).

¿Qué se puede deducir de la presente formalización?

- Cuáles instancias de Reemplazo se necesitan para poner a andar la maquinaria del forcing.
- Una verificación de que no se requiere Elección para hacerlo.

Esquema de reemplazo

"Si $x \mapsto F(x, \bar{x})$ es una función definible (usando parámetros $\bar{x}$), entonces para todo conjunto A, $F``A = \{F(a, \bar{x}) : a \in A\}$ es un conjunto.

Esquema de reemplazo

"Si $x \mapsto F(x, \bar{x})$ es una función definible (usando parámetros $\bar{x}$), entonces para todo conjunto A, $F``A = \{F(a, \bar{x}) : a \in A\}$ es un conjunto.

La triste realidad:

Esquema de reemplazo

"Si $x \mapsto F(x, \bar{x})$ es una función definible (usando parámetros $\bar{x}$), entonces para todo conjunto A, $F``A = \{F(a, \bar{x}) : a \in A\}$ es un conjunto.

La triste realidad:

Como una $\{\in\}$-fórmula

$$\forall A, \bar{x} : \Big(\forall y \in A\ \exists! z : \psi(y, z, \bar{x})\Big) \to \exists r : \forall w \Big(w \in r \longleftrightarrow \exists a \in A : \psi(a, w, \bar{x})\Big)$$

(Aquí, $w = F(a, \bar{x}) \iff \psi(a, w, \bar{x})$).

Esquema de reemplazo

"Si $x \mapsto F(x, \bar{x})$ es una función definible (usando parámetros $\bar{x}$), entonces para todo conjunto A, $F``A = \{F(a, \bar{x}) : a \in A\}$ es un conjunto.

La triste realidad:

Como una $\{\in\}$-fórmula

$$\forall A, \bar{x} : \big(\forall y \in A\ \exists! z : \psi(y, z, \bar{x})\big) \to \exists r : \forall w \big(w \in r \longleftrightarrow \exists a \in A : \psi(a, w, \bar{x})\big)$$

(Aquí, $w = F(a, \bar{x}) \iff \psi(a, w, \bar{x})$).
Luego, se complica obtener reemplazo a través de $G \circ F$ aún teniéndolo para G y F.

"Reemplazos Lambda"

Los reemplazos de la forma $x \mapsto \langle x, F(x, \bar{x})\rangle$ sí son componibles!

~~34~~ 22 instancias para dominarlas a todas

Construcciones básicas

- 2 instancias para clausura transitiva (iteración de $X \mapsto \bigcup X$ y absolutez).
- 1 instancia para definir $\in$-rango.
- 1 para la jerarquía acumulativa (V_α).

~~34~~ 22 instancias para dominarlas a todas

Construcciones básicas

- 2 instancias para clausura transitiva (iteración de $X \mapsto \bigcup X$ y absolutez).
- 1 instancia para definir $\in$-rango.
- 1 para la jerarquía acumulativa (V_α).

Aritmética cardinal

- 2 instancias para la definición de tipos de buen orden.
- 2 instancias para Aleph (reemplazo a través de $x \mapsto \text{tipo}(x)$ y la recursión).

~~34~~ 22 instancias para dominarlas a todas

También necesitamos una instancia extra ψ en M para que cada una de las φ anteriores valga en $M[G]$, donde $\psi(x, \alpha, y_1, \ldots, y_n)$ es

$$\alpha = \min\{\beta \mid \exists \tau \in V_\beta .\ snd(x) \Vdash \varphi\ [fst(x), \tau, y_1, \ldots, y_n]\}$$

Hasta ahora vamos $8 \cdot 2 = 16$.

~~34~~ 22 instancias para dominarlas a todas

También necesitamos una instancia extra ψ en M para que cada una de las φ anteriores valga en $M[G]$, donde $\psi(x, \alpha, y_1, \ldots, y_n)$ es

$$\alpha = \min\{\beta \mid \exists \tau \in V_\beta.\ snd(x) \Vdash \varphi\ [fst(x), \tau, y_1, \ldots, y_n]\}$$

Hasta ahora vamos $8 \cdot 2 = 16$.

Auxiliares de forcing

- 2 instancias para el Lema Δ (elección recursiva y su absolutez).
- 1 para la definición de "nombres" $\check{x}$ (incrustación $M \subseteq M[G]$).
- 1 para forzamiento de fórmulas atómicas.
- 1 para el nombre del filtro genérico ($x \mapsto \langle x, \check{x} \rangle$).
- 1 para absolutez de la recursión para ver $AC \to DC$.

No se requiere más “fuerza” para los Teoremas Fundamentales

Los Teoremas de Forcing no requieren instancias de Reemplazo extra en M.

No se requiere más “fuerza” para los Teoremas Fundamentales

Los Teoremas de Forcing no requieren instancias de Reemplazo extra en M.

Es decir, obtenemos

$$M[G] \models \varphi \iff \exists p \in G.\, p \Vdash \varphi \tag{1}$$

uniformemente en φ

No se requiere más “fuerza” para los Teoremas Fundamentales

Los Teoremas de Forcing no requieren instancias de Reemplazo extra en M.

Es decir, obtenemos

$$M[G] \models \varphi \iff \exists p \in G.\, p \Vdash \varphi \tag{1}$$

uniformemente en φ

Dependencia de Separación en φ

Las instancias de Separación requeridas por los Teoremas Fundamentales crecen recursivamente según la estructura de φ.

Isar

A continuación, un ejemplo (elegidazo!) de matemática formalizada, versus la versión en papel de Kunen [2011].

Theorem IV.2.27 *Let M be a ctm for ZF, let $\mathbb{P} \in M$ be a forcing poset, and let G be $\mathbb{P}$-generic over M. Then $M[G] \models ZF$. Furthermore, $M[G] \models ZFC$ if $M \models ZFC$.*

For Power Set (similarly to Union above), it is sufficient to prove that whenever $a \in M[G]$, there is a $b \in M[G]$ such that $\mathcal{P}(a) \cap M[G] \subseteq b$. Fix $\tau \in M^{\mathbb{P}}$ such that $\tau_G = a$. Let $Q = (\mathcal{P}(\mathrm{dom}(\tau) \times \mathbb{P}))^M$. This is the set of all names $\vartheta \in M^{\mathcal{P}}$ such that $\mathrm{dom}(\vartheta) \subseteq \mathrm{dom}(\tau)$. Let $\pi = Q \times \{\mathbb{1}\}$ and let $b = \pi_G = \{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$.

```
lemma Pow_inter_MG:
  assumes "a∈M[G]"
  shows "Pow(a) ∩ M[G] ∈ M[G]"
proof -
  from assms
  obtain τ where "τ ∈ M" "val(G, τ) = a"
    using GenExtD by auto
  let ?Q="Pow⇗M⇖(domain(τ)×ℙ)"
  let ?π="?Q×{1}"
  let ?b="val(G,?π)"
  from ‹τ∈M›
  have "domain(τ)×ℙ ∈ M" "domain(τ) ∈ M"
    by simp_all
  then
  have "?b ∈ M[G]"
    by (auto intro!:GenExtI)
  have "Pow(a) ∩ M[G] ⊆ ?b"
  proof
    fix c
    assume "c ∈ Pow(a) ∩ M[G]"
    then
    obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
      using GenExt_iff by auto
    let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
    have "arity(forces( ·0 ∈ 1· )) = 6"
      using arity_forces_at by auto
    with ‹domain(τ) ∈ M› ‹χ ∈ M›
    have "?ϑ ∈ M"
```

Theorem IV.2.27 *Let M be a ctm for ZF, let $\mathbb{P} \in M$ be a forcing poset, and let G be $\mathbb{P}$-generic over M. Then $M[G] \models ZF$. Furthermore, $M[G] \models ZFC$ if $M \models ZFC$.*

For Power Set (similarly to Union above), it is sufficient to prove that whenever $a \in M[G]$, there is a $b \in M[G]$ such that $\mathcal{P}(a) \cap M[G] \subseteq b$. Fix $\tau \in M^{\mathbb{P}}$ such that $\tau_G = a$. Let $Q = (\mathcal{P}(\mathrm{dom}(\tau) \times \mathbb{P}))^M$. This is the set of all names $\vartheta \in M^{\mathcal{P}}$ such that $\mathrm{dom}(\vartheta) \subseteq \mathrm{dom}(\tau)$. Let $\pi = Q \times \{\mathbb{1}\}$ and let $b = \pi_G = \{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$.

```
lemma Pow_inter_MG:
  assumes "a∈M[G]"
  shows "Pow(a) ∩ M[G] ∈ M[G]"
proof -
  from assms
  obtain τ where "τ ∈ M" "val(G, τ) = a"
    using GenExtD by auto
  let ?Q="Pow⇗M⇖(domain(τ)×ℙ)"
  let ?π="?Q×{1}"
  let ?b="val(G,?π)"
  from ‹τ∈M›
  have "domain(τ)×ℙ ∈ M" "domain(τ) ∈ M"
    by simp_all
  then
  have "?b ∈ M[G]"
    by (auto intro!:GenExtI)
  have "Pow(a) ∩ M[G] ⊆ ?b"
  proof
    fix c
    assume "c ∈ Pow(a) ∩ M[G]"
    then
    obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
      using GenExt_iff by auto
    let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
    have "arity(forces( ·0 ∈ 1· )) = 6"
      using arity_forces_at by auto
    with ‹domain(τ) ∈ M› ‹χ ∈ M›
    have "?ϑ ∈ M"
```

Theorem IV.2.27 *Let M be a ctm for ZF, let $\mathbb{P} \in M$ be a forcing poset, and let G be $\mathbb{P}$-generic over M. Then $M[G] \models ZF$. Furthermore, $M[G] \models ZFC$ if $M \models ZFC$.*

For Power Set (similarly to Union above), it is sufficient to prove that whenever $a \in M[G]$, there is a $b \in M[G]$ such that $\mathcal{P}(a) \cap M[G] \subseteq b$. Fix $\tau \in M^{\mathbb{P}}$ such that $\tau_G = a$. Let $Q = (\mathcal{P}(\mathrm{dom}(\tau) \times \mathbb{P}))^M$. This is the set of all names $\vartheta \in M^{\mathcal{P}}$ such that $\mathrm{dom}(\vartheta) \subseteq \mathrm{dom}(\tau)$. Let $\pi = Q \times \{\mathbb{1}\}$ and let $b = \pi_G = \{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$.

```
lemma Pow_inter_MG:
  assumes "a∈M[G]"
  shows "Pow(a) ∩ M[G] ∈ M[G]"
proof -
  from assms
  obtain τ where "τ ∈ M" "val(G, τ) = a"
    using GenExtD by auto
  let ?Q="Pow⇗M⇖(domain(τ)×ℙ)"
  let ?π="?Q×{1}"
  let ?b="val(G,?π)"
  from ‹τ∈M›
  have "domain(τ)×ℙ ∈ M" "domain(τ) ∈ M"
    by simp_all
  then
  have "?b ∈ M[G]"
    by (auto intro!:GenExtI)
  have "Pow(a) ∩ M[G] ⊆ ?b"
  proof
    fix c
    assume "c ∈ Pow(a) ∩ M[G]"
    then
    obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
      using GenExt_iff by auto
    let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
    have "arity(forces( ·0 ∈ 1· )) = 6"
      using arity_forces_at by auto
    with ‹domain(τ) ∈ M› ‹χ ∈ M›
    have "?ϑ ∈ M"
```

Theorem IV.2.27 *Let M be a ctm for ZF, let $\mathbb{P} \in M$ be a forcing poset, and let G be $\mathbb{P}$-generic over M. Then $M[G] \models ZF$. Furthermore, $M[G] \models ZFC$ if $M \models ZFC$.*

For Power Set (similarly to Union above), it is sufficient to prove that whenever $a \in M[G]$, there is a $b \in M[G]$ such that $\mathcal{P}(a) \cap M[G] \subseteq b$. Fix $\tau \in M^{\mathbb{P}}$ such that $\tau_G = a$. Let $Q = (\mathcal{P}(\mathrm{dom}(\tau) \times \mathbb{P}))^M$. This is the set of all names $\vartheta \in M^{\mathcal{P}}$ such that $\mathrm{dom}(\vartheta) \subseteq \mathrm{dom}(\tau)$. Let $\pi = Q \times \{\mathbb{1}\}$ and let $b = \pi_G = \{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$.

```
lemma Pow_inter_MG:
  assumes "a∈M[G]"
  shows "Pow(a) ∩ M[G] ∈ M[G]"
proof -
  from assms
  obtain τ where "τ ∈ M" "val(G, τ) = a"
    using GenExtD by auto
  let ?Q="Pow⇗M⇖(domain(τ)×ℙ)"
  let ?π="?Q×{1}"
  let ?b="val(G,?π)"
  from ‹τ∈M›
  have "domain(τ)×ℙ ∈ M" "domain(τ) ∈ M"
    by simp_all
  then
  have "?b ∈ M[G]"
    by (auto intro!:GenExtI)
  have "Pow(a) ∩ M[G] ⊆ ?b"
  proof
    fix c
    assume "c ∈ Pow(a) ∩ M[G]"
    then
    obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
      using GenExt_iff by auto
    let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
    have "arity(forces( ·0 ∈ 1· )) = 6"
      using arity_forces_at by auto
    with ‹domain(τ) ∈ M› ‹χ ∈ M›
    have "?ϑ ∈ M"
```

Theorem IV.2.27 *Let M be a ctm for ZF, let $\mathbb{P} \in M$ be a forcing poset, and let G be $\mathbb{P}$-generic over M. Then $M[G] \models ZF$. Furthermore, $M[G] \models ZFC$ if $M \models ZFC$.*

For Power Set (similarly to Union above), it is sufficient to prove that whenever $a \in M[G]$, there is a $b \in M[G]$ such that $\mathcal{P}(a) \cap M[G] \subseteq b$. Fix $\tau \in M^{\mathbb{P}}$ such that $\tau_G = a$. Let $Q = (\mathcal{P}(\mathrm{dom}(\tau) \times \mathbb{P}))^M$. This is the set of all names $\vartheta \in M^{\mathcal{P}}$ such that $\mathrm{dom}(\vartheta) \subseteq \mathrm{dom}(\tau)$. Let $\pi = Q \times \{\mathbb{1}\}$ and let $b = \pi_G = \{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$.

```
lemma Pow_inter_MG:
  assumes "a∈M[G]"
  shows "Pow(a) ∩ M[G] ∈ M[G]"
proof -
  from assms
  obtain τ where "τ ∈ M" "val(G, τ) = a"
    using GenExtD by auto
  let ?Q="Pow⇗M⇖(domain(τ)×ℙ)"
  let ?π="?Q×{1}"
  let ?b="val(G,?π)"
  from ‹τ∈M›
  have "domain(τ)×ℙ ∈ M" "domain(τ) ∈ M"
    by simp_all
  then
  have "?b ∈ M[G]"
    by (auto intro!:GenExtI)
  have "Pow(a) ∩ M[G] ⊆ ?b"
  proof
    fix c
    assume "c ∈ Pow(a) ∩ M[G]"
    then
    obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
      using GenExt_iff by auto
    let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
    have "arity(forces( ·0 ∈ 1· )) = 6"
      using arity_forces_at by auto
    with ‹domain(τ) ∈ M› ‹χ ∈ M›
    have "?ϑ ∈ M"
```

Theorem IV.2.27 *Let M be a ctm for ZF, let $\mathbb{P} \in M$ be a forcing poset, and let G be $\mathbb{P}$-generic over M. Then $M[G] \models ZF$. Furthermore, $M[G] \models ZFC$ if $M \models ZFC$.*

For Power Set (similarly to Union above), it is sufficient to prove that whenever $a \in M[G]$, there is a $b \in M[G]$ such that $\mathcal{P}(a) \cap M[G] \subseteq b$. Fix $\tau \in M^{\mathbb{P}}$ such that $\tau_G = a$. Let $Q = (\mathcal{P}(\mathrm{dom}(\tau) \times \mathbb{P}))^M$. This is the set of all names $\vartheta \in M^{\mathcal{P}}$ such that $\mathrm{dom}(\vartheta) \subseteq \mathrm{dom}(\tau)$. Let $\pi = Q \times \{\mathbb{1}\}$ and let $b = \pi_G = \{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$.

```
lemma Pow_inter_MG:
  assumes "a∈M[G]"
  shows "Pow(a) ∩ M[G] ∈ M[G]"
proof -
  from assms
  obtain τ where "τ ∈ M" "val(G, τ) = a"
    using GenExtD by auto
  let ?Q="Pow⇗M⇖(domain(τ)×ℙ)"
  let ?π="?Q×{1}"
  let ?b="val(G,?π)"
  from ‹τ∈M›
  have "domain(τ)×ℙ ∈ M" "domain(τ) ∈ M"
    by simp_all
  then
  have "?b ∈ M[G]"
    by (auto intro!:GenExtI)
  have "Pow(a) ∩ M[G] ⊆ ?b"
  proof
    fix c
    assume "c ∈ Pow(a) ∩ M[G]"
    then
    obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
      using GenExt_iff by auto
    let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
    have "arity(forces( ·0 ∈ 1· )) = 6"
      using arity_forces_at by auto
    with ‹domain(τ) ∈ M› ‹χ ∈ M›
    have "?ϑ ∈ M"
```

$\{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$. Fix $\varkappa \in M^{\mathbb{P}}$ such that $\varkappa_G = c$, and let $\vartheta = \{\langle \sigma, p\rangle : \sigma \in \mathrm{dom}(\tau) \wedge p \Vdash \sigma \in \varkappa\}$; $\vartheta \in M$ by the Definability Lemma. Since $\vartheta \in Q$, we are done if we can show that $\vartheta_G = c$. $\vartheta_G \subseteq c$ holds because $\vartheta_G = \{\sigma_G : \exists p \in G\ p \Vdash \sigma \in \varkappa\}$ and all these σ_G lie in $\varkappa_G = c$ by the definition of $\Vdash$. To prove $c \subseteq \vartheta_G$: since $c \subseteq a = \tau_G$, every element of c is of the form σ_G for some $\sigma \in \mathrm{dom}(\tau)$. Since $\sigma_G \in c = \varkappa_G$, apply the Truth Lemma and fix $p \in G$ such that $p \Vdash \sigma \in \varkappa$; then $\langle \sigma, p\rangle \in \vartheta$, so $\sigma_G \in \vartheta_G$.

```
have "?b ∈ M[G]"
  by (auto intro!:GenExtI)
have "Pow(a) ∩ M[G] ⊆ ?b"
proof
  fix c
  assume "c ∈ Pow(a) ∩ M[G]"
  then
  obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
    using GenExt_iff by auto
  let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
  have "arity(forces( ·0 ∈ 1· )) = 6"
    using arity_forces_at by auto
  with ‹domain(τ) ∈ M› ‹χ ∈ M›
  have "?ϑ ∈ M"
    using sats_fst_snd_in_M
    by simp
  with ‹domain(τ)×ℙ ∈ M›
  have "?ϑ ∈ ?Q"
    using Pow_rel_char by auto
  have "val(G,?ϑ) = c"
  proof(intro equalityI subsetI)
    fix x
    assume "x ∈ val(G,?ϑ)"
    then
    obtain σ p where 1: "⟨σ,p⟩∈?ϑ" "p∈G" "val(G,σ) =  x"
      using elem_of_val_pair
      by blast
    moreover from ‹⟨σ,p⟩∈?ϑ› ‹?ϑ ∈ M›
```

$\{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$. Fix $\varkappa \in M^{\mathbb{P}}$ such that $\varkappa_G = c$, and let $\vartheta = \{\langle \sigma, p\rangle : \sigma \in \mathrm{dom}(\tau) \wedge p \Vdash \sigma \in \varkappa\}$; $\vartheta \in M$ by the Definability Lemma. Since $\vartheta \in Q$, we are done if we can show that $\vartheta_G = c$. $\vartheta_G \subseteq c$ holds because $\vartheta_G = \{\sigma_G : \exists p \in G\ p \Vdash \sigma \in \varkappa\}$ and all these σ_G lie in $\varkappa_G = c$ by the definition of $\Vdash$. To prove $c \subseteq \vartheta_G$: since $c \subseteq a = \tau_G$, every element of c is of the form σ_G for some $\sigma \in \mathrm{dom}(\tau)$. Since $\sigma_G \in c = \varkappa_G$, apply the Truth Lemma and fix $p \in G$ such that $p \Vdash \sigma \in \varkappa$; then $\langle \sigma, p\rangle \in \vartheta$, so $\sigma_G \in \vartheta_G$.

```
have "?b ∈ M[G]"
  by (auto intro!:GenExtI)
have "Pow(a) ∩ M[G] ⊆ ?b"
proof
  fix c
  assume "c ∈ Pow(a) ∩ M[G]"
  then
  obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
    using GenExt_iff by auto
  let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
  have "arity(forces( ·0 ∈ 1· )) = 6"
    using arity_forces_at by auto
  with ‹domain(τ) ∈ M› ‹χ ∈ M›
  have "?ϑ ∈ M"
    using sats_fst_snd_in_M
    by simp
  with ‹domain(τ)×ℙ ∈ M›
  have "?ϑ ∈ ?Q"
    using Pow_rel_char by auto
  have "val(G,?ϑ) = c"
  proof(intro equalityI subsetI)
    fix x
    assume "x ∈ val(G,?ϑ)"
    then
    obtain σ p where 1: "⟨σ,p⟩∈?ϑ" "p∈G" "val(G,σ) =  x"
      using elem_of_val_pair
      by blast
    moreover from ‹⟨σ,p⟩∈?ϑ› ‹?ϑ ∈ M›
```

$\{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$. Fix $\varkappa \in M^{\mathbb{P}}$ such that $\varkappa_G = c$, and let $\vartheta = \{\langle \sigma, p\rangle : \sigma \in \mathrm{dom}(\tau) \wedge p \Vdash \sigma \in \varkappa\}$; $\vartheta \in M$ by the Definability Lemma. Since $\vartheta \in Q$, we are done if we can show that $\vartheta_G = c$. $\vartheta_G \subseteq c$ holds because $\vartheta_G = \{\sigma_G : \exists p \in G\ p \Vdash \sigma \in \varkappa\}$ and all these σ_G lie in $\varkappa_G = c$ by the definition of $\Vdash$. To prove $c \subseteq \vartheta_G$: since $c \subseteq a = \tau_G$, every element of c is of the form σ_G for some $\sigma \in \mathrm{dom}(\tau)$. Since $\sigma_G \in c = \varkappa_G$, apply the Truth Lemma and fix $p \in G$ such that $p \Vdash \sigma \in \varkappa$; then $\langle \sigma, p\rangle \in \vartheta$, so $\sigma_G \in \vartheta_G$.

```
have "?b ∈ M[G]"
  by (auto intro!:GenExtI)
have "Pow(a) ∩ M[G] ⊆ ?b"
proof
  fix c
  assume "c ∈ Pow(a) ∩ M[G]"
  then
  obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
    using GenExt_iff by auto
  let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
  have "arity(forces( ·0 ∈ 1· )) = 6"
    using arity_forces_at by auto
  with ‹domain(τ) ∈ M› ‹χ ∈ M›
  have "?ϑ ∈ M"
    using sats_fst_snd_in_M
    by simp
  with ‹domain(τ)×ℙ ∈ M›
  have "?ϑ ∈ ?Q"
    using Pow_rel_char by auto
  have "val(G,?ϑ) = c"
  proof(intro equalityI subsetI)
    fix x
    assume "x ∈ val(G,?ϑ)"
    then
    obtain σ p where 1: "⟨σ,p⟩∈?ϑ" "p∈G" "val(G,σ) =  x"
      using elem_of_val_pair
      by blast
    moreover from ‹⟨σ,p⟩∈?ϑ› ‹?ϑ ∈ M›
```

$\{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$. Fix $\varkappa \in M^{\mathbb{P}}$ such that $\varkappa_G = c$, and let $\vartheta = \{\langle \sigma, p\rangle : \sigma \in \mathrm{dom}(\tau) \wedge p \Vdash \sigma \in \varkappa\}$; $\vartheta \in M$ by the Definability Lemma. Since $\vartheta \in Q$, we are done if we can show that $\vartheta_G = c$. $\vartheta_G \subseteq c$ holds because $\vartheta_G = \{\sigma_G : \exists p \in G\ p \Vdash \sigma \in \varkappa\}$ and all these σ_G lie in $\varkappa_G = c$ by the definition of $\Vdash$. To prove $c \subseteq \vartheta_G$: since $c \subseteq a = \tau_G$, every element of c is of the form σ_G for some $\sigma \in \mathrm{dom}(\tau)$. Since $\sigma_G \in c = \varkappa_G$, apply the Truth Lemma and fix $p \in G$ such that $p \Vdash \sigma \in \varkappa$; then $\langle \sigma, p\rangle \in \vartheta$, so $\sigma_G \in \vartheta_G$.

```
have "?b ∈ M[G]"
  by (auto intro!:GenExtI)
have "Pow(a) ∩ M[G] ⊆ ?b"
proof
  fix c
  assume "c ∈ Pow(a) ∩ M[G]"
  then
  obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
    using GenExt_iff by auto
  let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
  have "arity(forces( ·0 ∈ 1· )) = 6"
    using arity_forces_at by auto
  with ‹domain(τ) ∈ M› ‹χ ∈ M›
  have "?ϑ ∈ M"
    using sats_fst_snd_in_M
    by simp
  with ‹domain(τ)×ℙ ∈ M›
  have "?ϑ ∈ ?Q"
    using Pow_rel_char by auto
  have "val(G,?ϑ) = c"
  proof(intro equalityI subsetI)
    fix x
    assume "x ∈ val(G,?ϑ)"
    then
    obtain σ p where 1: "⟨σ,p⟩∈?ϑ" "p∈G" "val(G,σ) =  x"
      using elem_of_val_pair
      by blast
    moreover from ‹⟨σ,p⟩∈?ϑ› ‹?ϑ ∈ M›
```

$\{\vartheta_G : \vartheta \in Q\}$. Now, consider any $c \in \mathcal{P}(a) \cap M[G]$; we need to show that $c \in b$. Fix $\varkappa \in M^{\mathbb{P}}$ such that $\varkappa_G = c$, and let $\vartheta = \{\langle \sigma, p\rangle : \sigma \in \mathrm{dom}(\tau) \wedge p \Vdash \sigma \in \varkappa\}$; $\vartheta \in M$ by the Definability Lemma. Since $\vartheta \in Q$, we are done if we can show that $\vartheta_G = c$. $\vartheta_G \subseteq c$ holds because $\vartheta_G = \{\sigma_G : \exists p \in G\ p \Vdash \sigma \in \varkappa\}$ and all these σ_G lie in $\varkappa_G = c$ by the definition of $\Vdash$. To prove $c \subseteq \vartheta_G$: since $c \subseteq a = \tau_G$, every element of c is of the form σ_G for some $\sigma \in \mathrm{dom}(\tau)$. Since $\sigma_G \in c = \varkappa_G$, apply the Truth Lemma and fix $p \in G$ such that $p \Vdash \sigma \in \varkappa$; then $\langle \sigma, p\rangle \in \vartheta$, so $\sigma_G \in \vartheta_G$.

```
have "?b ∈ M[G]"
  by (auto intro!:GenExtI)
have "Pow(a) ∩ M[G] ⊆ ?b"
proof
  fix c
  assume "c ∈ Pow(a) ∩ M[G]"
  then
  obtain χ where "c∈M[G]" "χ ∈ M" "val(G,χ) = c"
    using GenExt_iff by auto
  let ?ϑ="{⟨σ,p⟩ ∈domain(τ)×ℙ . p ⊩ ·0 ∈ 1· [σ,χ] }"
  have "arity(forces( ·0 ∈ 1· )) = 6"
    using arity_forces_at by auto
  with ‹domain(τ) ∈ M› ‹χ ∈ M›
  have "?ϑ ∈ M"
    using sats_fst_snd_in_M
    by simp
  with ‹domain(τ)×ℙ ∈ M›
  have "?ϑ ∈ ?Q"
    using Pow_rel_char by auto
  have "val(G,?ϑ) = c"
  proof(intro equalityI subsetI)
    fix x
    assume "x ∈ val(G,?ϑ)"
    then
    obtain σ p where 1: "⟨σ,p⟩∈?ϑ" "p∈G" "val(G,σ) =  x"
      using elem_of_val_pair
      by blast
    moreover from ‹⟨σ,p⟩∈?ϑ› ‹?ϑ ∈ M›
```

Otros enfoques a la Conjunteoría y al forcing

- Lean: Formalización completa de modelos a valores Booleanos e independencia de CH [Han and van Doorn, 2020].

Otros enfoques a la Conjunteoría y al forcing

- Lean: Formalización completa de modelos a valores Booleanos e independencia de CH [Han and van Doorn, 2020].
- Conjunteoría sobre Isabelle/HOL: `ZFC_in_HOL` [Paulson, 2019]

Otros enfoques a la Conjunteoría y al forcing

- Lean: Formalización completa de modelos a valores Booleanos e independencia de CH [Han and van Doorn, 2020].
- Conjunteoría sobre Isabelle/HOL: `ZFC_in_HOL` [Paulson, 2019]

Ojo al piojo: fuerza de consistencia

Isabelle/ZF	equiconsistente con ZF (?) [Paulson, 1989].
`ZFC_in_HOL`	aproximadamente ZF + 1 inaccesible.
Lean (CiC)	ZF + ω inaccesibles [Carneiro, 2019].

Mathporn (I)

$\texttt{forces}(0 \in 1) =$

Mathporn (I)

$$\texttt{forces}(0 \in 1) =$$

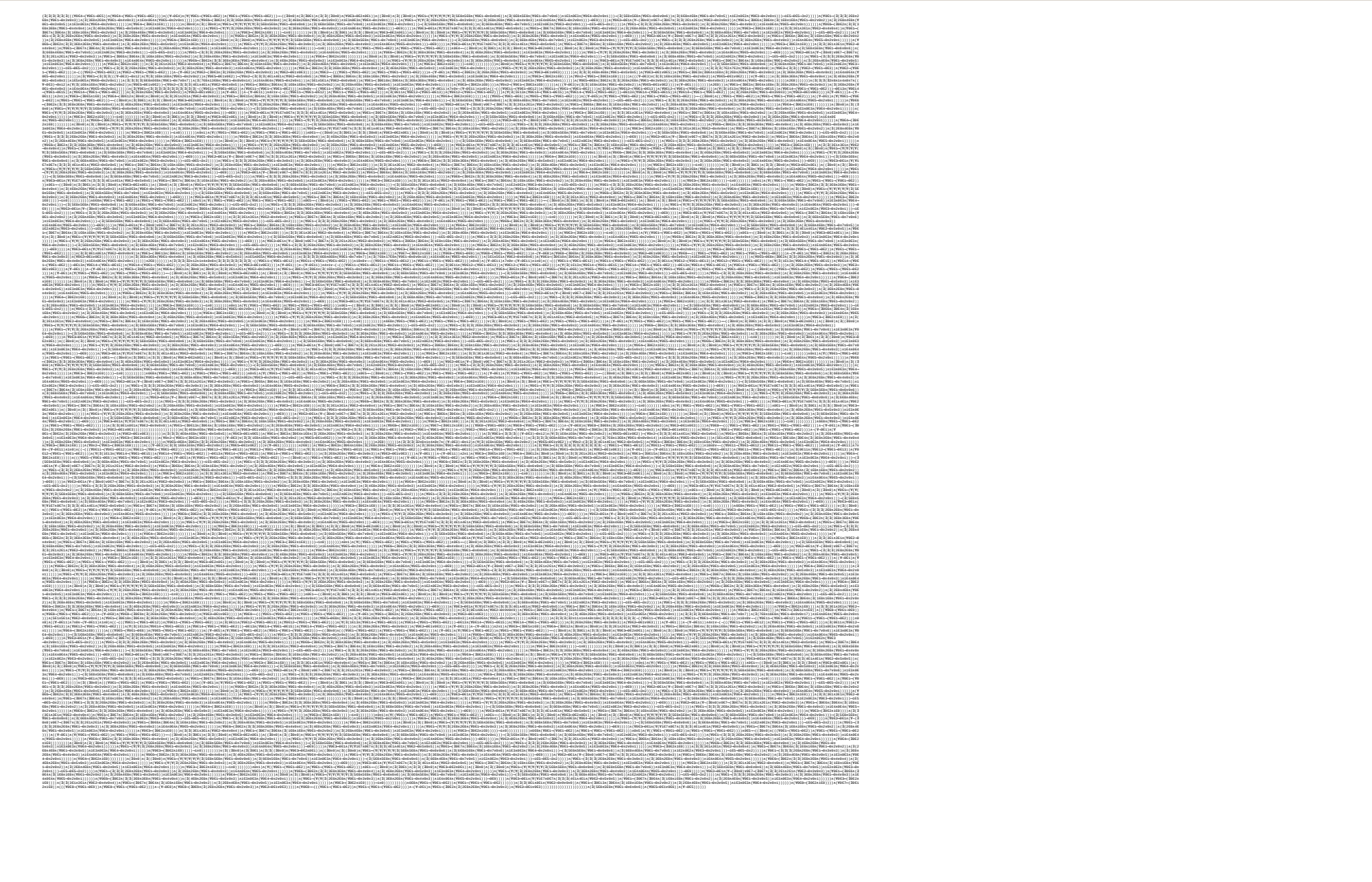
Mathporn (I)

$\texttt{forces}(0 \in 1) =$

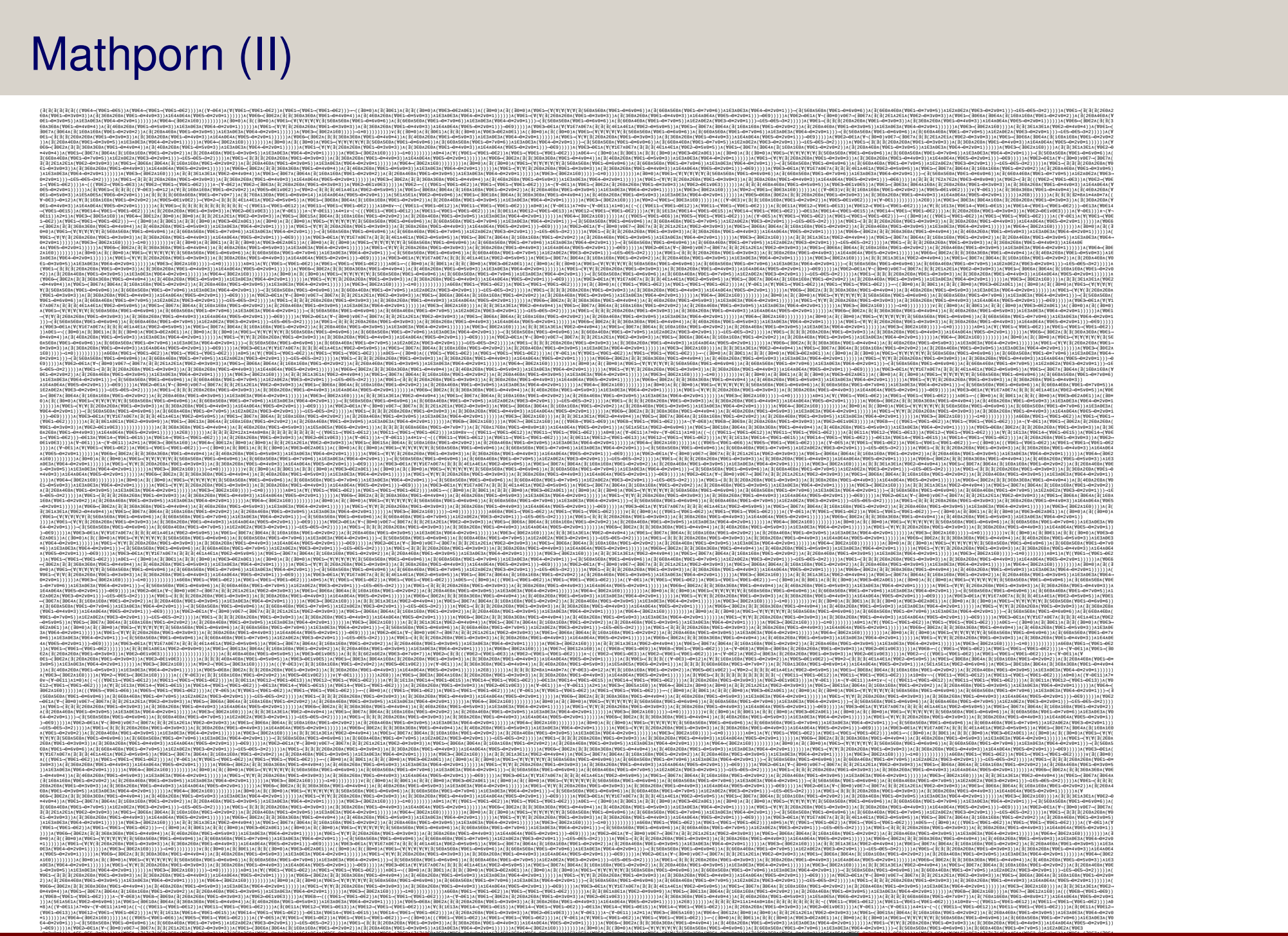
Y no mucho más…

$\texttt{forces}(\neg\neg(0 \in 1))$ no se puede expandir en Isabelle2021-1!

Mathporn (II)



Mathporn (II)

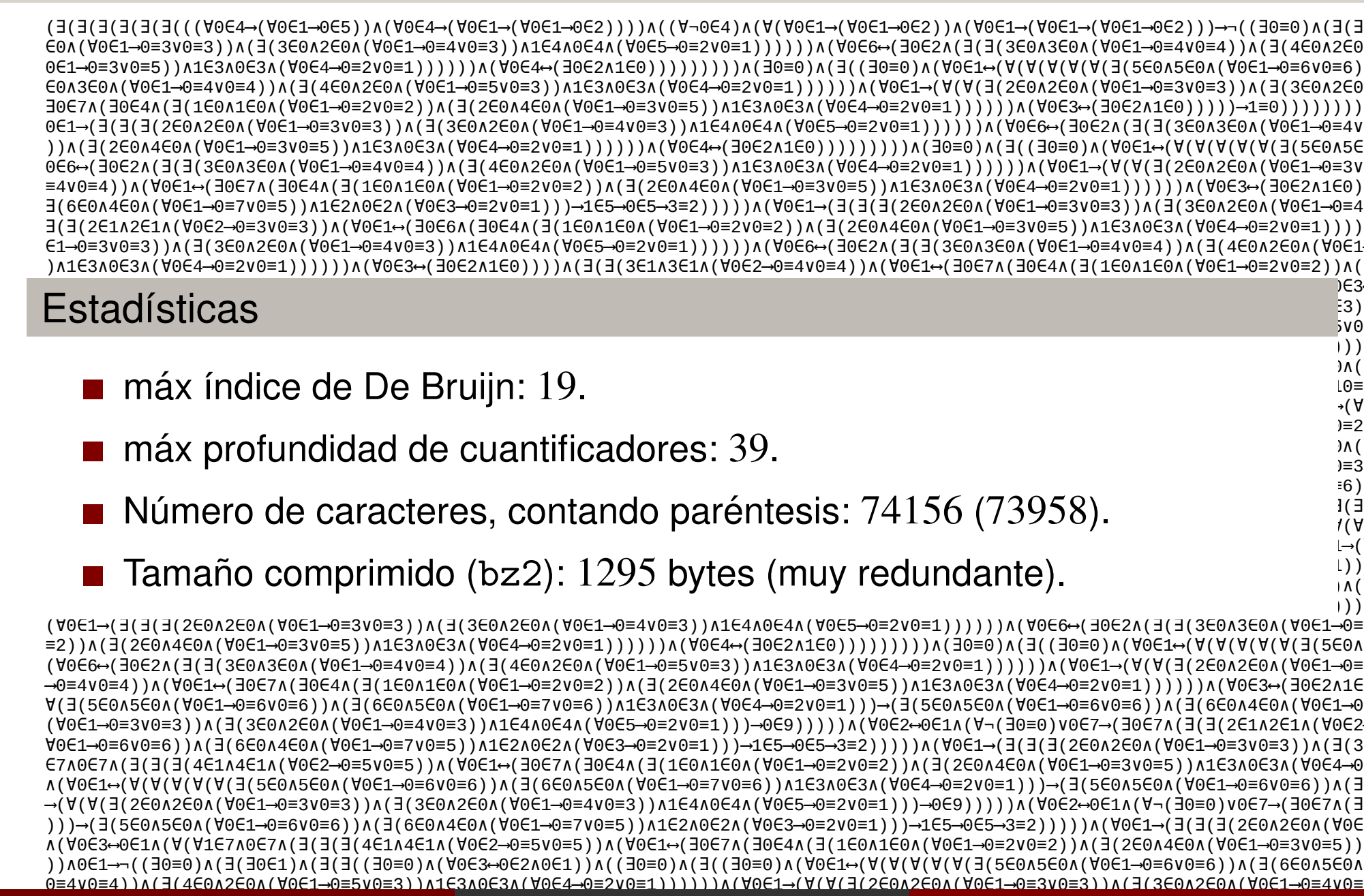


Mathporn (II)

Mathporn (II)

Mathporn (II): CH

Mathporn (II): CH



Estadísticas

- máx índice de De Bruijn: 19.
- máx profundidad de cuantificadores: 39.
- Número de caracteres, contando paréntesis: 74156 (73958).
- Tamaño comprimido (bz2): 1295 bytes (muy redundante).

Cómo leer nuestro trabajo

Algunos enlaces

- Entrada en la AFP:
  `https://devel.isa-afp.org/entries/Independence_CH.html`
- Sitio del proyecto: `https://cs.famaf.unc.edu.ar/~pedro/forcing/`
- Repositorio de código:
  `https://bitbucket.org/miguelpagano/forcing/src/master/src/`

Cómo leer nuestro trabajo

Algunos enlaces

- Entrada en la AFP:
  `https://devel.isa-afp.org/entries/Independence_CH.html`
- Sitio del proyecto: `https://cs.famaf.unc.edu.ar/~pedro/forcing/`
- Repositorio de código:
  `https://bitbucket.org/miguelpagano/forcing/src/master/src/`

Dónde arrancar

- Fíjense en el esquema que omite las pruebas. La última sección, Main Definitions. . . , tiene un resumen.
- Esa sección corresponde a la teoría `Definitions_Main`, desde la que se puede navegar al resto de los resultados.

Trabajo futuro

De formalización

- Construcción de ctms de ZFC a partir de un inaccesible;
- Forzamiento con posets de clases propias, usando un $\mathbb{P} \subseteq M$ definible en vez de $\mathbb{P} \in M$.
- Conectar este desarrollo con Isabelle/HOL.

Trabajo futuro

De formalización

- Construcción de ctms de ZFC a partir de un inaccesible;
- Forzamiento con posets de clases propias, usando un $\mathbb{P} \subseteq M$ definible en vez de $\mathbb{P} \in M$.
- Conectar este desarrollo con Isabelle/HOL.

Más minado meta-teórico

- Analizar Separación en más detalle instances in detail (operaciones de Gödel; junto a G. Figueroa).
- Continuar con el refinamiento de ZF-Constructible, en particular posponiendo los usos de Partes.

```
— ‹Kunen IV.3.5›
lemma ccc_fun_approximation_lemma:
  notes le_trans[trans]
  assumes "cccᴹ(P,leq)" "A∈M" "B∈M" "f∈M[G]" "f : A → B"
  shows
    "∃F∈M. F : A → Pow(B) ∧ (∀a∈A. f`a ∈ F`a ∧ |F`a|ᴹ ≤ ω)"
proof -
  from ‹f∈M[G]›
  obtain f_dot where "f = val(P,G,f_dot)" "f_dot∈M" using GenE
  with assms
  obtain p where "p ⊩ ·0:1→2· [f_dot, Aᵛ, Bᵛ]" "p∈G" "p∈M"
```

¡Gracias!

```
lemma Aleph2_extension_le_conti
  includes G_generic_lemmas
  shows "ℵ₂^M[G] ≤ 2^(↑ℵ₀^M[G],M[G])"
proof -
  have "ℵ₂^M ∈ M[G]" "Ord(ℵ₂^M)"
  moreover from this
  have "ℵ₂^M ≲^M[G] ω →^M[G] 2" [4
  moreover from calculation
  have "ℵ₂^M ≲^M[G] |ω →^M[G] 2|^M[G]
  ultimately
  have "|ℵ₂^M|^M[G] ≤ 2^(↑ℵ₀^M[G],M[G])" [
  then
  show "ℵ₂^M[G] ≤ 2^(↑ℵ₀^M[G],M[G])" [4
qed
```

References I

J. AVIGAD, Foundations, *arXiv e-prints* (2020). For the forthcoming Handbook of Proof Assistants and Their Applications in Mathematics and Computer Science, edited by Jasmin Blanchette and Assia Mahboubi.

M. CARNEIRO, "The Type Theory of Lean", Master's thesis, Carnegie Mellon University (2019).

P. COHEN, The independence of the continuum hypothesis, *Proc. Nat. Acad. Sci. U.S.A.* **50**: 1143–1148 (1963).

T. MATHLIB COMMUNITY, The lean mathematical library, in: Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2020, Association for Computing Machinery, New York, NY, USA: 367–381 (2020).

G. GONTHIER, Formal proof—the four-color theorem, *Notices Amer. Math. Soc.* **55**: 1382–1393 (2008).

G. GONTHIER, A. ASPERTI, J. AVIGAD, Y. BERTOT, C. COHEN, F. GARILLOT, S. LE ROUX, A. MAHBOUBI, R. O'CONNOR, S.O. BIHA, I. PASCA, L. RIDEAU, A. SOLOVYEV, E. TASSI, L. THÉRY, A machine-checked proof of the odd order theorem, in: Proceedings of the 4th International Conference on Interactive Theorem Proving, ITP'13, Springer-Verlag, Berlin, Heidelberg: 163–179 (2013).

T. HALES, M. ADAMS, G. BAUER, T.D. DANG, J. HARRISON, L.T. HOANG, C. KALISZYK, V. MAGRON, S. MCLAUGHLIN, T.T. NGUYEN, ET AL., A formal proof of the Kepler conjecture, *Forum Math. Pi* **5**: e2, 29 (2017).

J.M. HAN, F. VAN DOORN, A formal proof of the independence of the continuum hypothesis, in: J. Blanchette, C. Hritcu (Eds.), Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2020, New Orleans, LA, USA, January 20-21, 2020, ACM (2020).

References II

K. KUNEN, “Set Theory”, Studies in Logic, College Publications (2011), second edition. Revised edition, 2013.

L.C. PAULSON, The foundation of a generic theorem prover, *Journal of Automated Reasoning* **5**: 363–397 (1989).

L.C. PAULSON, The relative consistency of the axiom of choice mechanized using Isabelle/ZF, *LMS Journal of Computation and Mathematics* **6**: 198–248 (2003).

L.C. PAULSON, ALEXANDRIA: Large-scale formal proof for the working mathematician, Webpage, (2017 — accessed September 2018). EC Project: `https://bit.ly/2Nb26ys`.

L.C. PAULSON, Zermelo Fraenkel set theory in higher-order logic, *Archive of Formal Proofs* (2019). `http://isa-afp.org/entries/ZFC_in_HOL.html`, Formal proof development.

L.C. PAULSON, K. GRABCZEWSKI, Mechanizing set theory, *J. Autom. Reasoning* **17**: 291–323 (1996).

T. UNIVALENT FOUNDATIONS PROGRAM, “Homotopy Type Theory: Univalent Foundations of Mathematics”, `https://homotopytypetheory.org/book`, Institute for Advanced Study (2013).