An Analysis of Tennenbaum's Theorem in Constructive Type Theory

Marc Hermes and Dominik Kirst

Formal Structures for Computation and Deduction
August 3rd, 2022

SAARLAND UNIVERSITY
COMPUTER SCIENCE
SIC Saarland Informatics Campus

An Analysis of Tennenbaum's Theorem in Constructive Type Theory

An Analysis of Tennenbaum's Theorem in Constructive Type Theory

An Analysis of Tennenbaum's Theorem in Constructive Type Theory

Tennenbaum's Theorem

Usually stated as a "No-Go" Theorem[1]

> "There is no computable non-standard model of PA"

[1] Kaye (2011); Smith (2014)

Tennenbaum's Theorem

Usually stated as a “No-Go” Theorem[1]

> “There is no computable non-standard model of PA”

Framed positively it says:

> “Every computable model of PA is standard”

[1] Kaye (2011); Smith (2014)

Strangeness of Non-Standard Models

$\mathcal{M} \vDash$ PA is non-standard if there is $e : \mathcal{M}$: $\quad \forall n : \mathbb{N}.\ \overline{n} \neq e$

Strangeness of Non-Standard Models

$\mathcal{M} \vDash \mathsf{PA}$ is non-standard if there is $e : \mathcal{M}$: $\quad \forall n : \mathbb{N}.\ \overline{n} \neq e$

In PA we can define bounded products, like

$$\prod_{j<b} a_j$$

Strangeness of Non-Standard Models

$\mathcal{M} \vDash$ PA is non-standard if there is $e : \mathcal{M}$: $\quad \forall n : \mathbb{N}.\ \overline{n} \neq e$

In PA we can define bounded products, like

$$\prod_{j<b} a_j$$

b non-standard $\implies$ product ranges over all elements of $\mathbb{N}$ (and more)!

Strangeness of Non-Standard Models

$\mathcal{M} \vDash \mathsf{PA}$ is non-standard if there is $e : \mathcal{M}$: $\quad \forall n : \mathbb{N}.\ \overline{n} \neq e$

In PA we can define bounded products, like

$$\prod_{j<b} a_j$$

b non-standard $\implies$ product ranges over all elements of $\mathbb{N}$ (and more)!

Non-stdandard PA models contain "infinitary" computations.

Back to Tennenbaum

"There is no computable non-standard model of PA."

Back to Tennenbaum

> "There is no computable non-standard model of PA."

Computable Models

$$\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}$$

Computable Models

$$\begin{aligned} S^{\mathcal{M}} &: \mathbb{N} \to \mathbb{N} \\ +^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ \times^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \end{aligned}$$

Computable Models

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathbb{N} \to \mathbb{N} \\ +^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ \times^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \end{aligned}\right\}\ \text{need to be Turing computable.}$$

Computable Models

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathbb{N} \to \mathbb{N} \\ +^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ \times^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \end{aligned}\right\} \text{ need to be Turing computable.}$$

Definition (c.f. Smith (2014); Kaye (2011))

$\mathcal{M} \vDash \mathrm{PA}$ is called computable if its domain is $\mathbb{N}$ and $S^{\mathcal{M}}$, $+^{\mathcal{M}}$, $\times^{\mathcal{M}}$ are computable.

Computable Models

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathbb{N} \to \mathbb{N} \\ +^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ \times^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \end{aligned}\right\} \text{ need to be Turing computable.}$$

Definition (c.f. Smith (2014); Kaye (2011))

$\mathcal{M} \vDash \mathsf{PA}$ is called computable if its domain is $\mathbb{N}$ and $S^{\mathcal{M}}$, $+^{\mathcal{M}}$, $\times^{\mathcal{M}}$ are computable.

Furthermore assume :

Church's Thesis (CT)

Any *effectively computable* function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable.

… in Constructive Type Theory

In Constructive Type Theory

We can observe from the outside:

- only computable functions can be defined,
- no non-computable functions.

In Constructive Type Theory

We can observe from the outside:

- only computable functions can be defined,
- no non-computable functions.

No information about functions in a context:

"Assume there is a function $f : X \to Y \dots$"

Synthetic Computability

Every function $f : X \to Y$ is considered computable[2]

[2] Richman (1983); Bridges et al. (1987); Bauer (2006)

Synthetic Computability

Every function $f : X \to Y$ is considered computable[2]

Definition

Some notions from computability theory:

$$\begin{aligned}
\text{decidable}\, p &:= \exists f\ \forall(x : X).\, p\,x \leftrightarrow f\,x = \text{true} \\
\text{enumerable}\, p &:= \exists f\ \forall(x : X).\, p\,x \leftrightarrow \exists n.\, f\,n = \text{Some}\, x \\
\text{Discrete}\, X &:= \text{decidable}\, (\lambda(x, y) : X \times X.\, x = y) \\
\text{Enumerable}\, X &:= \text{enumerable}\, (\lambda(x : X).\top)
\end{aligned}$$

$$\text{MP} := \forall(f : \mathbb{N} \to \mathbb{N}).\, \neg\neg(\exists n.\, f\,n = 0) \to (\exists n.\, f\,n = 0)$$

[2]Richman (1983); Bridges et al. (1987); Bauer (2006)

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathbb{N} \to \mathbb{N} \\ +^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \\ \times^{\mathcal{M}} &: \mathbb{N} \times \mathbb{N} \to \mathbb{N} \end{aligned}\right\} \text{ need to be Turing computable.}$$

Definition

$\mathcal{M} \vDash \mathsf{PA}$ is called computable if its domain is $\mathbb{N}$ and $S^{\mathcal{M}}$, $+^{\mathcal{M}}$, $\times^{\mathcal{M}}$ are computable.

Furthermore assume :

Church's Thesis (CT)

Any *effectively computable* function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable.

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\} \text{ need to be Turing computable.}$$

Definition

$\mathcal{M} \vDash \mathsf{PA}$ is called computable if its domain is $\mathbb{N}$ and $S^{\mathcal{M}}$, $+^{\mathcal{M}}$, $\times^{\mathcal{M}}$ are computable.

Furthermore assume :

Church's Thesis (CT)

Any *effectively computable* function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable.

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\} \text{ need to be Turing computable.}$$

Definition

$\mathcal{M} \vDash \mathrm{PA}$ is called computable if its domain is $\mathbb{N}$ and $S^{\mathcal{M}}$, $+^{\mathcal{M}}$, $\times^{\mathcal{M}}$ are computable.

Furthermore assume :

Church's Thesis (CT)

Any *effectively computable* function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable .

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\}$$ ~~need to be Turing computable.~~

Definition

$\mathcal{M} \vDash$ PA is called computable if its domain is $\mathbb{N}$ and $S^{\mathcal{M}}$, $+^{\mathcal{M}}$, $\times^{\mathcal{M}}$ are computable.

Furthermore assume :

Church's Thesis (CT)

Any *effectively computable* function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable .

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\}$$ ~~need to be Turing computable.~~

Furthermore assume :

Church's Thesis (CT)

Any *effectively computable* function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable .

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\}$$ ~~need to be Turing computable.~~

Furthermore assume :

Church's Thesis (CT)

Any ~~*effectively computable*~~ function $f : \mathbb{N} \to \mathbb{N}$ is Turing computable .

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\}$$ ~~need to be Turing computable.~~

Furthermore assume :

Church's Thesis (CT)

Any ~~*effectively computable*~~ function $f : \mathbb{N} \to \mathbb{N}$ is ~~Turing computable~~ .

Computable Models (revised)

$$\left.\begin{aligned} S^{\mathcal{M}} &: \mathcal{M} \to \mathcal{M} \\ +^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \\ \times^{\mathcal{M}} &: \mathcal{M} \times \mathcal{M} \to \mathcal{M} \end{aligned}\right\}$$ ~~need to be Turing computable.~~

Furthermore assume :

Church's Thesis (CT)

Any function $f : \mathbb{N} \to \mathbb{N}$ is ???.

Church's Thesis for PA

Axiom ($\mathsf{CT_Q}$)[3]

For every $f : \mathbb{N} \to \mathbb{N}$ there is a Σ_1-formula φ_f s.t. for every $n : \mathbb{N}$,

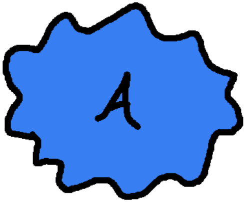
$$\mathsf{Q} \vdash \forall y.\, \varphi_f(\overline{n}, y) \leftrightarrow \overline{fn} = y$$

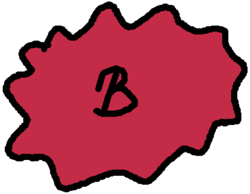
Can be added consistently to intuitionistic systems (e.g. constructive type theory[4]).

[3] $\mathsf{CT_Q}$ follows from more standard versions of CT.

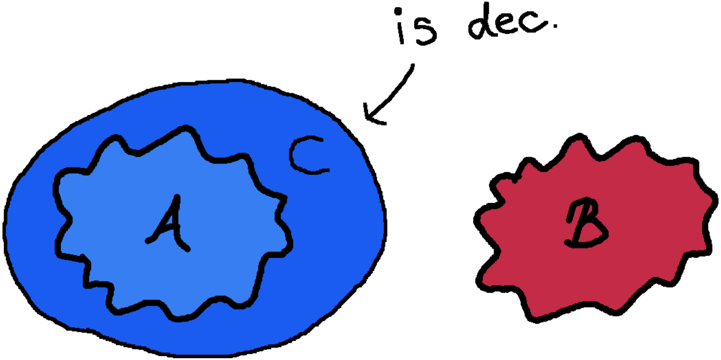
[4] Swan and Uemura (2019); Yamada (2020); Forster (2022)

Inseparable Sets





Inseparable Sets



Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By $\mathsf{CT_Q}$, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}. \overline{n} \neq e$.

- By $\mathsf{CT_Q}$, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.

- $\mathsf{Q} \vdash \alpha(\overline{n})$

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By CT_Q, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.
- $\mathsf{Q} \vdash \alpha(\overline{n}) \xrightarrow{\text{sound}} \mathcal{M} \vDash \alpha(\overline{n})$

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By $\mathsf{CT_Q}$, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.
- $\mathsf{Q} \vdash \alpha(\overline{n}) \xrightarrow{\text{sound}} \mathcal{M} \vDash \alpha(\overline{n}) \xrightarrow{\text{disj.}} \neg\, \mathcal{M} \vDash \beta(\overline{n})$

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By CT_Q, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.

- $\mathsf{Q} \vdash \alpha(\overline{n}) \xrightarrow{\text{sound}} \mathcal{M} \vDash \alpha(\overline{n}) \xrightarrow{\text{disj.}} \neg\, \mathcal{M} \vDash \beta(\overline{n}) \xrightarrow{\text{sound}} \neg\, \mathsf{Q} \vdash \beta(\overline{n})$

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By CT_Q, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.
- $\mathsf{Q} \vdash \alpha(\overline{\cdot}) \quad \subseteq \quad \mathcal{M} \vDash \alpha(\overline{\cdot}) \quad \subseteq \quad \neg\, \mathcal{M} \vDash \beta(\overline{\cdot}) \quad \subseteq \quad \neg\, \mathsf{Q} \vdash \beta(\overline{\cdot})$

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By $\mathsf{CT_Q}$, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.

- $\mathsf{Q} \vdash \alpha(\overline{\cdot}) \quad \subseteq \quad \underbrace{\mathcal{M} \vDash \alpha(\overline{\cdot})}_{\text{not decidable}} \quad \subseteq \quad \neg\, \mathcal{M} \vDash \beta(\overline{\cdot}) \quad \subseteq \quad \neg\, \mathsf{Q} \vdash \beta(\overline{\cdot})$

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By $\mathsf{CT_Q}$, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.
- $\mathsf{Q} \vdash \alpha(\overline{\cdot}) \quad \subseteq \quad \underbrace{\mathcal{M} \vDash \alpha(\overline{\cdot})}_{\text{not decidable}} \quad \subseteq \quad \neg\, \mathcal{M} \vDash \beta(\overline{\cdot}) \quad \subseteq \quad \neg\, \mathsf{Q} \vdash \beta(\overline{\cdot})$
- Let $c := \Pi_{(n<e \,\wedge\, \alpha(n))}\ \mathsf{prime}(n)$.

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By CT_Q, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.

- $\mathsf{Q} \vdash \alpha(\overline{\cdot}) \quad \subseteq \quad \underbrace{\mathcal{M} \vDash \alpha(\overline{\cdot})}_{\text{not decidable}} \quad \subseteq \quad \neg\, \mathcal{M} \vDash \beta(\overline{\cdot}) \quad \subseteq \quad \neg\, \mathsf{Q} \vdash \beta(\overline{\cdot})$

- Let $c := \Pi_{(n<e \,\wedge\, \alpha(n))}\ \mathsf{prime}(n)$.

- If $\lambda k.\, \overline{k} \mid c$ is decidable...

Proof Outline for Tennenbaum

Assume there is $e : \mathcal{M}$ such that $\forall n : \mathbb{N}.\, \overline{n} \neq e$.

- By $\mathsf{CT_Q}$, there are $\mathsf{Q} \vdash \alpha(\overline{\cdot})$, $\mathsf{Q} \vdash \beta(\overline{\cdot})$ which are inseparable.
- $\mathsf{Q} \vdash \alpha(\overline{\cdot}) \quad \subseteq \quad \underbrace{\mathcal{M} \vDash \alpha(\overline{\cdot})}_{\text{not decidable}} \quad \subseteq \quad \neg\, \mathcal{M} \vDash \beta(\overline{\cdot}) \quad \subseteq \quad \neg\, \mathsf{Q} \vdash \beta(\overline{\cdot})$
- Let $c := \Pi_{(n<e \,\wedge\, \alpha(n))}\ \mathsf{prime}(n)$.
- If $\lambda k.\, \overline{k} \mid c$ is decidable...
- then so is $\mathcal{M} \vDash \alpha(\overline{\cdot})$ (Contradiction!)

Proof Outline for Tennenbaum

$$\mathcal{M} \text{ is non-standard} \implies \exists c : \mathcal{M} \text{ s.t. } \overline{\cdot} \mid c \text{ is not decidable.}$$

Analysis of Proofs

[5] Also implies that the model is discrete.

Analysis of Proofs

$$\mathcal{M} \cong \mathbb{N} := \forall e : \mathcal{M}\, \exists n : \mathbb{N}.\, \overline{n} = e$$
$$\mathcal{M} > \mathbb{N} := \exists e : \mathcal{M}\, \forall n : \mathbb{N}.\, \overline{n} \neq e$$

Boolos et al. (2002)	MP	Discrete $\mathcal{M}$	Enumerable $\mathcal{M} \leftrightarrow \mathcal{M} \cong \mathbb{N}$
Kaye (2011); Smith (2014)	MP	Discrete $\mathcal{M}$	$(\forall d : \mathcal{M}.\, \mathsf{Dec}(\overline{\cdot} \mid d)) \leftrightarrow \mathcal{M} \cong \mathbb{N}$

[5] Also implies that the model is discrete.

Analysis of Proofs

$$\mathcal{M} \cong \mathbb{N} := \forall e : \mathcal{M}\, \exists n : \mathbb{N}.\, \overline{n} = e$$
$$\mathcal{M} > \mathbb{N} := \exists e : \mathcal{M}\, \forall n : \mathbb{N}.\, \overline{n} \neq e$$

Boolos et al. (2002)	MP	Discrete $\mathcal{M}$	Enumerable $\mathcal{M} \leftrightarrow \mathcal{M} \cong \mathbb{N}$
Kaye (2011); Smith (2014)	MP	Discrete $\mathcal{M}$	$(\forall d : \mathcal{M}.\, \mathsf{Dec}(\overline{\cdot} \mid d)) \leftrightarrow \mathcal{M} \cong \mathbb{N}$
Makholm (2014)			$(\exists d : \mathcal{M}.\, \neg\mathsf{Dec}(\overline{\cdot} \mid d)) \leftrightarrow \mathcal{M} > \mathbb{N}$

[5] Also implies that the model is discrete.

Analysis of Proofs

$$\mathcal{M} \cong \mathbb{N} := \forall e : \mathcal{M}\, \exists n : \mathbb{N}.\, \overline{n} = e$$
$$\mathcal{M} > \mathbb{N} := \exists e : \mathcal{M}\, \forall n : \mathbb{N}.\, \overline{n} \neq e$$

Boolos et al. (2002)	MP	Discrete $\mathcal{M}$	Enumerable $\mathcal{M} \leftrightarrow \mathcal{M} \cong \mathbb{N}$
Kaye (2011); Smith (2014)	MP	Discrete $\mathcal{M}$	$(\forall d : \mathcal{M}.\, \mathsf{Dec}(\overline{\cdot} \mid d)) \leftrightarrow \mathcal{M} \cong \mathbb{N}$
Makholm (2014)			$(\exists d : \mathcal{M}.\, \neg\mathsf{Dec}(\overline{\cdot} \mid d)) \leftrightarrow \mathcal{M} > \mathbb{N}$
McCarty (1987, 1988)	MP	Constr. Sem.[5]	$\forall \mathcal{M}.\, \mathcal{M} \cong \mathbb{N}$

[5] Also implies that the model is discrete.

Future Work

- Investigate and mechanize other proofs, try to prove the additional assumptions into e.g. Makholm.
- Interpret $S, +, \times$ as functional relations and make separate computability assumptions. Eliminate dependance on $\mathsf{CT_Q}$.
- Mechanize proof that computability of $\times^{\mathcal{M}}$ suffices. (McAloon, 1982)
- HA categorical $\implies$ MP?

Take-Home Messages

Take-Home Messages

Constructively, PA has no non-standard models.

Take-Home Messages

Constructively, PA has no non-standard models.

Usage of CT makes formalization and proofs elegant.
(Concise, but precise!)

`https://www.ps.uni-saarland.de/extras/tennenbaum/`

Bibliography I

Bauer, A. (2006). First steps in synthetic computability theory. *Electronic Notes in Theoretical Computer Science*, 155:5–31.

Boolos, G. S., Burgess, J. P., and Jeffrey, R. C. (2002). *Computability and logic*. Cambridge university press.

Bridges, D., Richman, F., et al. (1987). *Varieties of constructive mathematics*, volume 97. Cambridge University Press.

Forster, Y. (2022). Parametric Church's Thesis: Synthetic Computability Without Choice. In Artemov, S. and Nerode, A., editors, *Logical Foundations of Computer Science*, pages 70–89, Cham. Springer International Publishing.

Kaye, R. (2011). Tennenbaum's theorem for models of arithmetic. *Set Theory, Arithmetic, and Foundations of Mathematics. Ed. by J. Kennedy and R. Kossak. Lecture Notes in Logic. Cambridge*, pages 66–79.

Bibliography II

Kirst, D. and Hermes, M. (2021). Synthetic Undecidability and Incompleteness of First-Order Axiom Systems in Coq. In Cohen, L. and Kaliszyk, C., editors, *12th International Conference on Interactive Theorem Proving (ITP 2021)*, volume 193 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 23:1–23:20, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Kirst, D., Hostert, J., Dudenhefner, A., Forster, Y., Hermes, M., Koch, M., Larchey-Wendling, D., Mück, N., Peters, B., Smolka, G., and Wehr, D. (2022). A Coq library for mechanised first-order logic. In *The Coq Workshop*.

Kleene, S. C. (1945). On the interpretation of intuitionistic number theory. *The journal of symbolic logic*, 10(4):109–124.

Makholm, H. (2014). Tennenbaum's theorem without overspill. Mathematics Stack Exchange. (version: 2014-01-24).

McAloon, K. (1982). On the complexity of models of arithmetic. *The Journal of Symbolic Logic*, 47(2):403–415.

McCarty, D. C. (1987). Variations on a thesis: intuitionism and computability. *Notre Dame Journal of Formal Logic*, 28(4):536–580.

Bibliography III

McCarty, D. C. (1988). Constructive validity is nonarithmetic. *The Journal of Symbolic Logic*, 53(4):1036–1041.

Plisko, V. E. (1990). Constructive formalization of the Tennenbaum theorem and its applications. *Mathematical notes of the Academy of Sciences of the USSR*, 48(3):950–957.

Richman, F. (1983). Church's thesis without tears. *The Journal of symbolic logic*, 48(3):797–803.

Smith, P. (2014). Tennenbaum's theorem. Technical report.

Swan, A. and Uemura, T. (2019). On Church's Thesis in Cubical Assemblies.

T. Godziszewski, M. and Hamkins, J. D. (2017). Computable quotient presentations of models of arithmetic and set theory. *arXiv e-prints*, pages arXiv–1702.

Tennenbaum, S. (1959). Non-archimedean models for arithmetic. *Notices of the American Mathematical Society*, 6(270):44.

Yamada, N. (2020). Game semantics of Martin-Löf type theory, part III: its consistency with Church's thesis.

Backup : A Forest of Tennenbaum proofs

- The original: Tennenbaum (1959)
- Via inseparable sets: Smith (2014); Kaye (2011)
- Via inseparable sets (refined): Makholm (2014)
- Diagonal proof: Boolos et al. (2002)
- Constructively: McCarty (1988, 1987); Plisko (1990)
- Assuming only computability of $\times^{\mathcal{M}}$: McAloon (1982)
- Tennenbaum phenomena: T. Godziszewski and Hamkins (2017)

Backup : Coq Development

- Formalization of FOL based on Kirst et al. (2022); Kirst and Hermes (2021)
- 3000 loc reused from the above.
- 1320 loc for the mechanization of all variants of Tennenbaum.

html documentation of the code:
`https://www.ps.uni-saarland.de/extras/tennenbaum`

The repository can be found at:
`https://github.com/HermesMarc/Tennenbaum-CTT`

Backup : Church's Thesis

- There are consistency proofs for CT in univalent type theory (Swan and Uemura, 2019) and Martin-Löf type theory (Yamada, 2020).
- CT together with MP is consistent in HA (Kleene, 1945) and univalent type theory (Swan and Uemura, 2019).
- CT_Q can be derived from CT for μ-recursive functions or equivalent models.

Backup : Additional Assumptions

For the proof of Makholm's and McCarty's results, the following two assumptions were also made:

Todo 1

There are Σ_1 formulas $\alpha(x), \beta(x)$ such that Q $\vdash \alpha(\overline{\cdot})$, Q $\vdash \beta(\overline{\cdot})$ are inseparable and HA $\vdash \neg\exists x.\, \alpha(x) \land \beta(x)$.

Todo 2

For any Δ_1 formula $\varphi(x, y)$ we have

$$\mathsf{HA} \vdash \forall n b\, \exists c\, \forall u < n.\, (\exists x < b. \varphi(x, u)) \leftrightarrow \Pi(u) \mid c$$

Both of them are expected to be provable and therefore removable as assumptions.

Backup: Existence of Non-Standard Models

Compactness Theorem

Assuming LEM, if every finite $\Gamma \subseteq \mathcal{T}$ has a model, then so has $\mathcal{T}$.

Backup: Existence of Non-Standard Models

Compactness Theorem

Assuming LEM, if every finite $\Gamma \subseteq \mathcal{T}$ has a model, then so has $\mathcal{T}$.

Expand PA to PA* by a constant c and axioms $c \neq 0,\ c \neq 1,\ c \neq 2,\ c \neq 3\ \ldots$

Backup: Existence of Non-Standard Models

Compactness Theorem

Assuming LEM, if every finite $\Gamma \subseteq \mathcal{T}$ has a model, then so has $\mathcal{T}$.

Expand PA to PA* by a constant c and axioms $c \neq 0,\ c \neq 1,\ c \neq 2,\ c \neq 3 \ldots$

For any finite $\Gamma = \{\ldots, c \neq k_1, \ldots, c \neq k_n\} \subseteq$ PA*

Backup: Existence of Non-Standard Models

Compactness Theorem

Assuming LEM, if every finite $\Gamma \subseteq \mathcal{T}$ has a model, then so has $\mathcal{T}$.

Expand PA to PA* by a constant c and axioms $c \neq 0,\ c \neq 1,\ c \neq 2,\ c \neq 3 \ldots$

For any finite $\Gamma = \{\ldots, c \neq k_1, \ldots, c \neq k_n\} \subseteq$ PA*

extending $\mathbb{N}$ by $c^{\mathbb{N}} := 1 + k_1 + \ldots + k_n$ we get $\mathbb{N} \vDash \Gamma$

Backup: Existence of Non-Standard Models

Compactness Theorem

Assuming LEM, if every finite $\Gamma \subseteq \mathcal{T}$ has a model, then so has $\mathcal{T}$.

Expand PA to PA* by a constant c and axioms $c \neq 0,\ c \neq 1,\ c \neq 2,\ c \neq 3\ \ldots$

For any finite $\Gamma = \{\ldots, c \neq k_1, \ldots, c \neq k_n\} \subseteq$ PA*

extending $\mathbb{N}$ by $\quad c^{\mathbb{N}} := 1 + k_1 + \ldots + k_n \quad$ we get $\quad \mathbb{N} \vDash \Gamma$

By compactness PA* has a model $\mathcal{M}$, where

$$\forall n \in \mathbb{N}. \quad \overline{n} \neq c^{\mathcal{M}}$$