# Memo : Tennenbaum's Theorem in Constructive Type Theory

## 1 Preliminaries

We start by giving Tennenbaum's Theorem as e.g. formulated in [5]

**Theorem 1.1 (Tennenbaum):** *If $M \equiv \langle \mathbb{N}, 0, \oplus, \otimes, = \rangle$ is a* $\mathsf{PA}$ *model which is not isomorphic to the standard model* $\mathbb{N}$ *then neither* $\oplus$ *nor* $\otimes$ *can be recursive.*

The above formulation only considers models whose domain can be identified with $\mathbb{N}$ and are thus countable. In general we can certainly restrict our attention to countable models, since no uncountable model can be recursive.

In this memo we follow the proof of Tennenbaum's Theorem as presented in [5] but derive the result in a constructive type theory (CTT) providing types for the natural numbers $\mathbb{N}$, booleans $\mathbb{B}$ and propositions $\mathbb{P}$. We start out be fixing some definitions on the level of our meta-theory:

**Definition 1.2:**

- $\mathsf{reflect}\,(P : \mathbb{P})(b : \mathbb{B}) := P \leftrightarrow b = \mathsf{true}$
- $\mathsf{Dec}\ \ (p : X \to \mathbb{P}) := \exists\, f : X \to \mathbb{B}\ \forall\, x.\ \mathsf{reflect}\ p(x)\ f(x)$
- $\mathsf{Enum}\ \ (p : \mathbb{N} \to \mathbb{P}) := \exists\, f : \mathbb{N} \to \mathbb{N}\ \ \forall\, x.\ p(x) \leftrightarrow \exists\, n.\ f(n) = S(x)$
- $\mathsf{Stable}\ \ (p : X \to \mathbb{P}) := \forall\, x, \neg\neg p(x) \to p(x)$

Any type $X$ is a **data type** iff there is an enumerator of $X$ and $X$ has decidable equality.

There are three different theories of arithmetic we will be considering, namely $\mathsf{PA}$, $\mathsf{HA}$ and $\mathsf{Q}$. As a reminder: $\mathsf{Q}$ is $\mathsf{PA}$ without the induction scheme but with the added axiom $\forall\, x.\, x = 0 \vee \exists\, y.\, Sy = x$ and $\mathsf{HA}$ has the same axioms as $\mathsf{PA}$ but works with an intuitionistic deduction system.

Most importantly, throughout the whole text we will assume

**Proposition 1.3 (Church's Thesis ($\mathsf{CT}$)):** *Every function $f : \mathbb{N} \to \mathbb{N}$ in our constructive meta-theory is $\mu$-recursive.*

and for the purpose of conciseness:

**Proposition 1.4:** *There is an injective function $\pi : \mathbb{N} \to \mathbb{N}$ only producing prime numbers.*

Using $\mathsf{CT}$ we can extend a standard result for $\mu$-recursive functions ([4] Thm.39.2.) to all functions of our meta theory:

**Definition 1.5:**

$$\mathsf{CT_Q} := \forall f : \mathbb{N} \to \mathbb{N}\ \exists \varphi_f.\ \Sigma_1\, \varphi_f\ \ \wedge\ \ \forall n.\ \mathsf{Q} \vdash \forall x, \varphi_f(\overline{n}, x) \leftrightarrow \overline{f(n)} = x$$

giving us the ability to represent any meta-level function on the object-level.

We can already make use of this in the following proposition:

**Proposition 1.6:** *There are inseparable r.e. formulas, meaning a pair of unary $\Sigma_1$ formulas $\alpha, \beta$ in the language of* PA *such that $\forall n.\ \mathbb{N} \vDash \neg(\alpha \wedge \beta)(n)$ and for any decidable predicate $p : \mathbb{N} \to \mathbb{P}$ we can not have*

$$\forall n.\ \mathbb{N} \vDash \alpha(n) \to p(n) \quad \textit{and} \quad \forall n.\ \neg\big(p(n) \wedge \mathbb{N} \vDash \beta(n)\big).$$

**Proof:** Let $\psi_n$ enumerate all formulas in PA, then we define the disjoint predicates $A(n) := \mathsf{PA} \vdash \neg\psi_n(n)$ and $B(n) := \mathsf{PA} \vdash \psi_n(n)$. One can show that $A, B$ are enumerable.

Given any enumerable predicate $p$ with function $f$ such that $p(x) \leftrightarrow \exists n.f(n) = S(x)$, we get a $\Sigma_1$ formula $\varphi_f$ representing $f$ by using $\mathsf{CT_Q}$. One can now check that $p(n) \leftrightarrow \mathbb{N} \vDash \exists n.\varphi_f(n, S(x))$. Using this Idea on $A$ and $B$, we get candidate $\Sigma_1$ formulas $\alpha, \beta$.

Given any decidable predicate $p$, we can similarly to above get a $\Sigma_1$ formula $\varphi_p$ representing $p$ in Q, and deduce $p(n) \leftrightarrow \mathbb{N} \vDash \varphi_p(n)$. By the enumeration of formulas, there is $c : \mathbb{N}$ such that $\psi_c = \varphi_p$. We are now left to show that

$$\forall n.\ \mathbb{N} \vDash \psi_n(n) \to \mathbb{N} \vDash \psi_c(n) \quad \text{and} \quad \forall n.\ \neg\big(\mathbb{N} \vDash \neg\psi_c(n) \wedge \mathbb{N} \vDash \psi_n(n)\big).$$

always leads to a contradiction. This will follow from a case analysis on $p(c) \vee \neg p(c)$. □

# 2 HA models

We now gather some statements which are true in any HA model $M$.

We can define an embedding $\nu : \mathbb{N} \to M$ by $\nu(0) := 0$, $\nu(n+1) := S(\nu(n))$ which can easily be shown to be injective. Image points of $\nu$ will be called **standard numbers** or simply **numerals** and we will use the notation $\hat{n} := \nu(n)$ for them.

**Lemma 2.1:** *For any binary $\Delta_0$ formula $\varphi$ we have*

$$M \vDash \forall x\, y.\ (\exists z < x.\ \varphi(z, y))\ \vee\ \neg(\exists z < x.\ \varphi(z, y)).$$

**Proof:** By induction on the $\Delta_0$ formula $\varphi$. □

Wo now come to one of the most important ideas we need on our way to Tennenbaum's Theorem. Informally; when given any finite set $L$ of natural numbers, we can code this set by a single number $c$ in the following way

$$c := \text{product of the prime numbers } \pi(u) \text{ for all } u \in L$$

and the decoding is possible due to the prime factorization theorem. Given any decidable predicate $p$ and bound $n : \mathbb{N}$ we can now use this idea to code the finite set $\{u < n \mid p(u)\}$. The following Lemma shows the existence of the desired code number in $\mathbb{N}$, together with its defining properties.

**Lemma 2.2:** *If the predicate $p : \mathbb{N} \to \mathbb{P}$ is propositionally decidable, then for every bound $n : \mathbb{N}$ there is a code $c : \mathbb{N}$ such that*

$$\forall u. \big(u < n \to p(u) \leftrightarrow \pi(u) \mid c\big) \wedge \big(\pi(u) \mid c \to u < n\big).$$

**Proof:** We do a proof by induction on $n$. For $n = 0$ we can choose $c = 1$. For the successor case: if $\neg p(n)$ we can simply take the code $c$ given by the induction hypothesis, otherwise if $p(n)$ we multiply the given $c$ with $\pi(n)$. In both cases the conditions are checked by making use of the prime property as well as the injectivity of $\pi$. □

Using $\mathsf{CT_Q}$ we get a formula $\varphi_\pi$ representing $\pi$, making it possible to show a slightly weaker version of the above lemma, establishing coding of finite sets in $M$.

**Lemma 2.3:** *For any binary $\Delta_0$ formula $\varphi$ and $n : \mathbb{N}$ we have*

$$M \vDash \forall b\, \exists c\, \forall u < \hat{n}.\ (\exists z < b.\ \varphi(z, u)) \ \leftrightarrow\ \exists p.\ \varphi_\pi(u, p) \wedge p \mid c.$$

**Proof:** By Lemma 2.1, the predicate $\lambda n.\ M \vDash \exists z < b.\ \varphi(z, \hat{n})$ is propositionally decidable. One then checks that $\hat{c}$ with the $c : \mathbb{N}$ given by Lemma 2.2 proves the existence claim. Of great importance is the fact that $u < \hat{n}$. It tells us that $u$ is a numeral, reducing the goal to checking the equivalence for numerals $u$ only, which is guaranteed by Lemma 2.2. □

**Remark 2.4:** If $u$ can be shown to be a numeral, we can use the defining property of $\varphi_\pi$ given by $\mathsf{CT_Q}$ to show that $\widehat{\pi(u)} \mid c \ \leftrightarrow\ M \vDash \exists p.\, \varphi_\pi(u, p) \wedge p \mid c$. In the following we will therefore slightly abuse notation and write $\pi(u) \mid c$ for $\exists p.\, \varphi_\pi(u, p) \wedge p \mid c$.

# 3 Tennenbaum in CTT

Again let $M \vDash \mathsf{HA}$ , then we have a predicate expressing that an element is a standard number $\mathsf{std}(e) := \exists n.\ \hat{n} = e$ and we will call $M$ a **standard model** iff $\forall e. \mathsf{std}(e)$ (i.e. iff $\nu$ is surjective).

**Lemma 3.1:** *$M$ is a standard model iff there is a unary formula $\varphi$ with*

$$\forall e.\ \big(\mathsf{std}(e)\ \leftrightarrow\ M \vDash \varphi(e)\big).$$

**Proof:** If $M$ is a standard model, then the formula $x = x$ shows the claim.

Given a formula $\varphi$ with the desired property, we certainly have $\varphi(\hat{0})$ since $\hat{0}$ is a numeral, and $\varphi(x) \leftrightarrow \mathsf{std}(x)$ clearly implies $\mathsf{std}(Sx)$ and hence $\varphi(Sx)$. Thus by induction in the model, we have $\forall x. M \vDash \varphi(x)$ which is equivalent to $\forall x. \mathsf{std}(x)$.□

**Corollary 3.2 (Overspill):** *Assume that $M$ in not standard and* Stable std*. If $\varphi$ is any unary formula with $M \vDash \varphi(\hat{n})$ for all $n : \mathbb{N}$ then $\neg\neg\exists\, e : M. \neg\mathsf{std}(e) \wedge M \vDash \varphi(e)$.*

**Proof:** Assume that $\neg\exists\, e : M. \neg\mathsf{std}(e) \wedge M \vDash \varphi(e)$. By the stability of std this implies $\forall\, e : M. M \vDash \varphi(e) \to \mathsf{std}(e)$. Since $M$ is not standard, the assumption that $\varphi$ holds on all numerals together with Lemma 3.1 gives us $\neg\forall\, e. M \vDash \varphi(e) \to \mathsf{std}(e)$ and thus a contradiction. □

Using Overspill we will now be able to remove the restriction of only being able to code finite sets in $M$.

**Lemma 3.3:** *If $M$ is not standard and* Stable std*, then for any binary $\Delta_0$ formula $\varphi$ we have*

$$\neg\neg\forall\,(b : M)\,\exists\,(c : M)\,\forall\,(n : \mathbb{N}).\ (\exists\, x < b.\ \varphi(x, \hat{n})) \ \leftrightarrow\ \pi(n) \mid c.$$

**Proof:** Recall that by Lemma 2.3 we have for all $n : \mathbb{N}$

$$M \vDash \forall\, b\,\exists\, c\,\forall\, u < \hat{n}.\ (\exists\, z < b.\ \varphi(z, u)) \ \leftrightarrow\ \pi(u) \mid c$$

And thus by applying Overspill we get

$$\neg\neg\exists\, e.\ \mathsf{std}(e)\ \wedge\ M \vDash \forall\, b\,\exists\, c\,\forall\, u < e.\ (\exists\, z < b.\ \varphi(z, u)) \ \leftrightarrow\ \pi(u) \mid c.$$

Since we are trying to show a double-negated goal, we can now get rid of the double negations in the goal and the assumption. So we have $e : M$ non-standard such that for every $b : M$ there is $c : M$ with

$$M \vDash \forall\, u < e.\ (\exists\, z < b.\ \varphi(z, u)) \ \leftrightarrow\ \pi(u) \mid c.$$

In particular, if $u$ is equal to any numeral $\hat{n}$ we have $\hat{n} < e$ and thus

$$M \vDash (\exists\, z < b.\ \varphi(z, \hat{n})) \ \leftrightarrow\ \widehat{\pi(n)} \mid c.$$ □

**Lemma 3.4:** *If $M$ is a data type, then for all $0 < n$ and $e : M$ we can compute a boolean value $b : \mathbb{B}$ such that* reflect $(\hat{n} \mid e)$ $b$*.*

**Proof:** By Euclid's lemma we have unique $r : \mathbb{N}$, $d : M$ with

$$e = d \cdot \hat{n} + \hat{r}\ \wedge\ r < n$$

If $r = 0$, we have $e = d \cdot \hat{n}$ and thus clearly reflect $(\hat{n} \mid e)$ true.

Now consider $r \neq 0$. If we had $\hat{n} \mid e$ then $e = d' \cdot \hat{n} + 0$ for some $d' : M$, and by the uniqueness part of Euclid $r = 0$, giving us a contradiction and hence reflect $(\hat{n} \mid e)$ false □

**Remark 3.5:** Since the above proof hides the reason we need $M$ to be a data type, let us mention it here.

The Euclidean lemma only gives the propositional existence of $r : \mathbb{N}$ and $d : M$. We are then not allowed to access them for the proof since the goal is asking for a boolean decision. If $M$ has decidable equality, the expression $e = d \cdot \hat{n} + \hat{r} \wedge r < n$ is decidable and if $M$ can be enumerated, there will be some $k : \mathbb{N}$ which will hit $d$ during the enumeration. The problem then comes down to searching for $r, k : \mathbb{N}$ which satisfy the mentioned decidable predicate. Since Euclid guarantees us that the search will end, it shows that we can get $r, k$ computationally.

**Lemma 3.6:** *If $M$ is not standard then $\neg\neg\,\exists\, c.\ \neg\mathsf{Dec}(\lambda n.\, \widehat{\pi(n)} \mid c)$.*

**Proof:** Let $\alpha', \beta'$ be inseparable $\Sigma_1$ formulas, which are equivalent to formulas $\exists\, z.\alpha(z, x)$, $\exists\, z.\beta(z, x)$ where $\alpha, \beta$ are binary $\Delta_0$ formulas. Since $\alpha', \beta'$ are disjoint we can show

$$\mathbb{N} \vDash \forall\, x, y, z < \hat{n}\, \neg(\alpha(y, x) \wedge \beta(z, x))$$

for every $n : \mathbb{N}$. By soundness and the decidability of $\Delta_0$ formulas we then get

$$M \vDash \forall\, x, y, z < \hat{n}\, \neg(\alpha(y, x) \wedge \beta(z, x)).$$

Using Overspill we (under a double negation) get $e : M$ with

$$M \vDash \forall\, x, y, z < e\, \neg(\alpha(y, x) \wedge \beta(z, x))$$

which guarantees disjointness of $\alpha, \beta$ when everything is bounded by $e$.

We now define the predicate $X := \lambda n.\, M \vDash \exists\, z < e.\ \alpha(z, \hat{n})$ and note that

- If $\mathbb{N} \vDash \exists\, z.\alpha(z, n)$ there is $m : \mathbb{N}$ with $\mathbb{N} \vDash \alpha(m, n)$. By $\Delta_0$ completeness and soundness this gives us $M \vDash \alpha(\hat{m}, \hat{n})$ which finally implies $X(n)$.
- Assume that $X(n) \wedge \mathbb{N} \vDash \exists\, z.\beta(z, \hat{n})$. Then similarly to above, there is $m : \mathbb{N}$ with $M \vDash \beta(\hat{m}, \hat{n})$, showing $M \vDash \exists\, z < e.\beta(z, \hat{n})$. Together with $X(n)$ this contradicts the disjointness of $\alpha, \beta$ under the bound $e$.

This shows that $X$ can not be decidable due to the inseparability of $\alpha'$ and $\beta'$. By Lemma 3.3 however, there is a code $c : M$ such that $X(n) \leftrightarrow \pi(n) \mid c$ which by Lemma 3.4 means that $X$ is decidable. Hence we have a contradiction. ☐

**Theorem 3.7 (Tennenbaum):** *Assuming $\mathsf{CT_Q}$ then for any data type $M \vDash \mathsf{HA}$ if $\mathsf{Stable\ std}$ then $\forall\, e.\mathsf{std}(e)$, telling us that $\nu : \mathbb{N} \to M$ is an isomorphism.*

**Proof:** By the stability of $\mathsf{std}$ our goal is equivalent to $\neg\neg\forall\, e.\mathsf{std}(e)$, so for the sake of contradiction, assume that $M$ is not standard. Then by Lemma 3.6 (we can remove the $\neg\neg$ since we are trying to show $\bot$) there is an $e : M$ with $\neg\mathsf{Dec}(\lambda n.\, \widehat{\pi(n)} \mid e)$, contradicting however the result of Lemma 3.3. ☐

# 4 Proof by McCarty

We will now present another constructive proof that is due to Charles McCarty [2] [3]. For the proof to work one can either assume unique choice in the form

**Definition 4.1:** $\mathsf{UC} := \forall R.\ \forall x. \exists! y. R(x,y) \rightarrow \exists (f : \mathbb{N} \rightarrow \mathbb{B}).\ \forall x. R(x, f(x))$

or situate the models in a constructive setting (e.g. in intuitionistic $\mathsf{ZF}$), assuring that the interpretation of the disjunction $\vee$ is a computational decision. We also requite a stronger notion of inseparability, namely:

**Definition 4.2:** Inseparable r.e. formulas $\alpha, \beta$ are called **object inseparable** iff $\mathsf{HA} \vdash \neg \exists x. \alpha(x) \wedge \beta(x)$.

**Theorem 4.3:** *Given* $\mathsf{UC}$ *and object inseparable formulas, then for any* $M \vDash \mathsf{HA}$ *we have* $\forall e.\ \neg\neg \mathsf{std}(e)$*. By further assuming Markov's Principle we get* $\forall e.\ \mathsf{std}(e)$*.*

**Proof:** Assume that $M$ has a non-standard element $e : M$ and let $\alpha$, $\beta$ the inseparable $\Sigma_1$ formulas. For any unary formula $\varphi$ it is possible to show

$$\mathsf{HA} \vdash \forall x \neg\neg \forall y < x.\ \varphi(y) \vee \neg\varphi(y).$$

Using soundness and instantiating the above for $\alpha$ and $e$, we get

$$M \vDash \neg\neg \forall y < e.\ \alpha(y) \vee \neg\alpha(y).$$

We are trying to prove a contradiction, so we can get rid of the $\neg\neg$ and since any numeral $\hat{n}$ is smaller then the non-standard number $e$ we get $M \vDash \alpha(\hat{n})\ \vee\ \neg M \vDash \alpha(\hat{n})$ which can then be used to show

$$\forall n\ \exists! b.\ \mathsf{reflect}\ (\lambda n.\, M \vDash \alpha(\hat{n}))\ b$$

Thus by $\mathsf{UC}$ there is a decider for $\lambda n.\, M \vDash \alpha(\hat{n})$ and we have

- $\mathbb{N} \vDash \alpha(n) \rightarrow \mathsf{HA} \vdash \alpha(n) \rightarrow M \vDash \alpha(\hat{n})$ by $\Sigma_1$ completeness and soundness.
- Assuming $M \vDash \alpha(\hat{n}) \wedge \mathbb{N} \vDash \beta(n)$ we get $M \vDash \beta(\hat{n})$ similarly to above. Together with $M \vDash \alpha(\hat{n})$ this contradicts $\mathsf{HA} \vdash \neg \exists x. \alpha(x) \wedge \beta(x)$.

This shows that $X$ would be separating $\alpha$ and $\beta$, giving us the desired contradiction. □

# 5 Discussion

We will now analyse the assumptions that go into the proof of Theorem 3.7.

(1) It should be possible to explicitly construct a formula $\varphi_\pi$ representing $\pi$ in $\mathsf{Q}$, which would remove one usage of $\mathsf{CT_Q}$ in the proof.

(2) We note that the need for stability of $\mathsf{std}$ in Section 3 comes from the usage of Lemma 3.6 where the dependence goes back to Overspill. It turns out that redoing the proof using object inseparable formulas, one can completely avoid the usage of Overspill. This was sketched in [1] and is also at play in the proof by McCarty. It should again be possible to explicitly construct object inseparable formulas, eliminating the last dependance on $\mathsf{CT_Q}$, which would enable us to drop $\mathsf{CT}$ as an assumption.

(3) The assumption that $M$ needs to be a data type can be weakened. Since we have $\mathsf{HA} \vdash \forall\, x\, y.\, x = y \lor x \neq y$ it suffices to have decidable apartness. It is also possible to do a different proof of Lemma 3.4 assuming a witness operator for $M$, instead of an enumerator.

Overall these observations would get us

**Proposition 5.1:** *If $M$ has an enumerator or witness operator and decidable apartness, we have $\forall\, e. \neg\neg \mathsf{std}(e)$. Further assuming Markov's Principle then shows $\forall\, e. \mathsf{std}(e)$.*

We will now turn to comparing the results Theorem 3.7 and Theorem 4.3. The main difference, looked at from the standpoint of CTT, lies in the definition of $\vDash$. In the Coq development mechanizing the proof as presented in Section 3, the predicate $\vDash$ is placed into the impredicative type of propositions $\mathbb{P}$. For an alternative treatment of McCarty's one can put $\vDash$ into $\mathbb{T}$, turning the interpretation of $\lor$ into an informative sum type, making it possible to drop the assumption $\mathsf{UC}$. Under these circumstances we get decidable equality of the type as a theorem from $\mathsf{HA} \vdash \forall\, x\, y.\, x = y \lor x \neq y$ and soundness. It does not seem possible to internally express the enumerability of $M$ in $\mathsf{HA}$, showing that Theorem 4.3 is less restrictive when it comes to the type $M$ of the model.

The current Coq development treats both addition and multiplication as function symbols in the logic, meaning the interpretation of both symbols in the model correspond are recursive functions. To really put the development in line with the original statement of Tennenbaum's theorem, this needs to be changed in a future version; capturing one of the functions via a relation in the signature, and therefore no longer enforcing it to be recursive. It is also important to note that the current development directly interprets the object equality as model equality, which should and possibly can be weakened.

Both Section 3 and Theorem 4.3 only need the stability of $\mathsf{std}$ to establish $\forall\, e. \mathsf{std}(e)$, leading to the interesting question wether something close to

$$\text{Markov's Principle} \longleftrightarrow \forall\, M.\ \mathsf{Stable}\ \mathsf{sdt}_M$$

can be shown in any sensible way.

# Bibliography

[1] Henning Makholm (https://math.stackexchange.com/users/14366/hmakholm-left-over-monica). *Tennenbaums theorem without overspill*. Mathematics Stack Exchange. (version: 2014-01-24). URL: https://math.stackexchange.com/q/649457.

[2] Charles McCarty. "Constructive Validity is Nonarithmetic". In: *The Journal of Symbolic Logic* 53.4 (1988), pp. 1036–1041. ISSN: 00224812. URL: http://www.jstor.org/stable/2274603.

[3] Charles McCarty. "Variations on a thesis: intuitionism and computability." In: *Notre Dame Journal of Formal Logic* 28.4 (1987), pp. 536–580.

[4] Peter Smith. *An introduction to Gödel's theorems*. Cambridge University Press, 2013.

[5] Peter Smith. "Tennenbaum's Theorem". In: (2014).