The HoTT Perspective

Numbers as Inductive Types up to Path Equivalence

YonedaAI Research
Univalent Correspondence Project
Magneton Foundational Studies

3 May 2026

Abstract

We give a self-contained development of *Homotopy Type Theory* (HoTT) as the synthetic language in which natural numbers, real numbers, and the transcendentals π and e admit an encoding-free description. The natural numbers are the inductive type $\mathbb{N}$ generated by $\mathsf{zero} : \mathbb{N}$ and $\mathsf{succ} : \mathbb{N} \to \mathbb{N}$; the integer 58 is the canonical term $\mathsf{succ}^{58}\,\mathsf{zero}$. Identity types $a =_A b$ are the central novelty: rather than mere propositions, they carry the homotopical content of *path spaces*. We develop path induction, transport, and coherence, then state Voevodsky's univalence axiom $(A = B) \simeq (A \simeq B)$ and prove its fundamental consequence: the type $\sum_{X:\mathcal{U}}(X \simeq \mathbb{N})$ of "structures equivalent to $\mathbb{N}$" is contractible, so every natural number object is canonically equal as a type. We treat higher inductive types, illustrated by the circle S^1 with $\pi_1(S^1) = \mathbb{Z}$, and Booij's higher inductive–inductive construction of the Cauchy reals as an h-set, inside which π and e are defined as the centres of contractible types of solutions to characteristic structural conditions, formalising "unique existence." Cubical type theory (Cohen–Coquand–Huber–Mörtberg) makes univalence computational, while two-level, modal, and directed HoTT (Riehl–Shulman) extend the framework to synthetic $(\infty, 1)$-category theory. We close by identifying HoTT as the internal language of $(\infty, 1)$-toposes, making the categorical and structural views of arithmetic literally co-extensive. A companion Haskell implementation models paths, refl, $\mathsf{transport}$, sym and trans at the type level via GADTs.

Contents

1 Introduction

In the previous papers of this series we asked "What is 58?" and answered it on four levels: as a numeral (Paper 01), as a von Neumann ordinal (Paper 02), as the value of a unique morphism out of a Natural Number Object (Paper 03), and as the representable presheaf $\mathrm{Hom}(-, 58)$ (Paper 04). Each refinement traded ontological substance for relational structure. The question that survives Paper 04 is whether the categorical viewpoint is *itself* an encoding – whether "58 as the representable $\mathrm{Hom}(-, 58)$" replaces one substantialist commitment with another.

Homotopy type theory answers "no" constructively. In HoTT every type is already identified up to equivalence *by axiom*, so the question "which set, presheaf, or initial object is the real $\mathbb{N}$?" is dissolved into the slogan

equivalent things are equal.

Two natural number objects, two presentations of the reals, two $K(G, 1)$'s for the same group are not merely isomorphic up to a chosen comparison map – they are paths in the universe.

This paper develops the perspective in detail. We restrict ourselves to the *Book HoTT* of [1] except where noted, and present the material in standard Martin-Löf style with one universe $\mathcal{U}$ for clarity; the ramified hierarchy $\mathcal{U}_0 : \mathcal{U}_1 : \mathcal{U}_2 : \cdots$ is needed for foundational coherence but plays no role in the arithmetic content.

1.1 Roadmap

Section 2 fixes notation for dependent type theory. Section 3 defines $\mathbb{N}$ as an inductive type and recovers primitive recursion. Section 4 introduces identity types, J, transport, and the path groupoid. Section 5 states the univalence axiom and proves the contractibility of the space of NNO-equivalents. Section 6 treats higher inductive types: the circle, $\pi_1(S^1) = \mathbb{Z}$, propositional truncation, the Cauchy reals. Section 7 sketches cubical type theory. Section 8 surveys two-level, modal, and directed HoTT. Section 9 places π and e inside the Cauchy reals. Section 10 identifies HoTT as the internal language of $(\infty, 1)$-toposes. Section 11–15 record the principal results and discuss limitations.

1.2 Notational conventions

We write $a : A$ for "a is a term of type A," $f : A \to B$ for the type of (non-dependent) functions, $\prod_{x:A} B(x)$ for the dependent product ("Π-type"), and $\sum_{x:A} B(x)$ for the dependent sum ("Σ-type"). The identity type of $a, b : A$ is written $a =_A b$ or $\mathsf{Id}_A(a, b)$. The constant path is $\mathsf{refl}_a : a =_A a$. Equivalence is $A \simeq B$. We use $\mathcal{U}$ for "the" universe. We deliberately distinguish three kinds of "equality":

- the judgmental / definitional equality (a meta-level notion), written $a \equiv b$, used in computation rules;
- the propositional / identity-type equality, written $a = b$, an inhabitant of the type $\mathsf{Id}_A(a, b)$;
- type equivalence, written $A \simeq B$, an inhabitant of $\sum_{f:A\to B} \mathsf{isEquiv}(f)$.

Univalence asserts that the canonical map relating the second and third (at the universe level) is itself an equivalence.

2 Mathematical Framework

We assume Martin-Löf's intensional dependent type theory MLTT_ω as the substrate [2]. The pertinent rules are summarized below informally; for the formal calculus see [1, Appendix A].

2.1 Dependent products and sums

For a type family $B : A \to \mathcal{U}$, the Π-type $\prod_{x:A} B(x)$ has introduction by λ-abstraction and elimination by application. The Σ-type $\sum_{x:A} B(x)$ has introduction (a, b) for $a : A, b : B(a)$ and elimination by projections fst and snd together with a dependent eliminator ind_Σ.

Definition 2.1 (Function type)**.** $A \to B$ abbreviates $\prod_{x:A} B$ when B does not depend on x.

Definition 2.2 (Cartesian product)**.** $A \times B$ abbreviates $\sum_{x:A} B$ when B does not depend on x.

2.2 The unit, empty, and sum types

The unit type $\mathbf{1}$ has a single constructor $\star : \mathbf{1}$ and the eliminator "everything depending on $\mathbf{1}$ is determined by its value at $\star$." The empty type $\mathbf{0}$ has no constructors and the eliminator $\mathsf{ind}_{\mathbf{0}} : \prod_{C:\mathbf{0}\to\mathcal{U}} \prod_{z:\mathbf{0}} C(z)$ (ex falso). The sum type $A + B$ has $\mathsf{inl} : A \to A + B$ and $\mathsf{inr} : B \to A + B$.

2.3 Universes and Russell vs. Tarski

We adopt *Russell-style* universes: $\mathcal{U}_0 : \mathcal{U}_1 : \cdots$ with $\mathcal{U}_i \subset \mathcal{U}_{i+1}$ and a term of $\mathcal{U}$ is itself a type. All "$\mathcal{U}$"s in this paper denote a generic universe; statements are true at every level.

3 Natural Numbers as an Inductive Type

3.1 The inductive definition

Definition 3.1 (Natural numbers)**.** The type $\mathbb{N} : \mathcal{U}$ is generated by the formation rule $\mathbb{N} : \mathcal{U}$ and the two constructors

$$\begin{aligned} \mathsf{zero} &: \mathbb{N}, \\ \mathsf{succ} &: \mathbb{N} \to \mathbb{N}. \end{aligned}$$

The dependent eliminator (*induction principle*) is

$$\mathsf{ind}_{\mathbb{N}} : \prod_{P:\mathbb{N}\to\mathcal{U}} P(\mathsf{zero}) \to \Big(\prod_{n:\mathbb{N}} P(n) \to P(\mathsf{succ}\, n)\Big) \to \prod_{n:\mathbb{N}} P(n).$$

The computation rules are $\mathsf{ind}_{\mathbb{N}}\, P\, c_0\, c_s\, \mathsf{zero} \equiv c_0$ and $\mathsf{ind}_{\mathbb{N}}\, P\, c_0\, c_s\, (\mathsf{succ}\, n) \equiv c_s\, n\, (\mathsf{ind}_{\mathbb{N}}\, P\, c_0\, c_s\, n)$.

Definition 3.2 (Numerals)**.** Define $\overline{n} : \mathbb{N}$ recursively by $\overline{0} := \mathsf{zero}$ and $\overline{n+1} := \mathsf{succ}\, \overline{n}$. In particular,

$$\overline{58} = \mathsf{succ}(\mathsf{succ}(\mathsf{succ}(\cdots(\mathsf{succ}\, \mathsf{zero})\cdots))) \quad \text{(58 applications of } \mathsf{succ}\text{)}.$$

3.2 Recursion and arithmetic

The non-dependent recursor

$$\mathsf{rec}_{\mathbb{N}} : \prod_{C:\mathcal{U}} C \to (\mathbb{N} \to C \to C) \to \mathbb{N} \to C$$

is the special case where $P(n) := C$ for all n.

Example 3.3 (Addition)**.** Define $+ : \mathbb{N} \to \mathbb{N} \to \mathbb{N}$ by $m + n := \mathsf{rec}_{\mathbb{N}}\, \mathbb{N}\, n\, (\lambda k.\, \mathsf{succ})\, m$, equivalently by $\mathsf{zero} + n \equiv n$ and $(\mathsf{succ}\, m) + n \equiv \mathsf{succ}(m + n)$. Multiplication and exponentiation are defined analogously.

Proposition 3.4 (NNO universal property, internal)**.** *For any type $C : \mathcal{U}$, point $c_0 : C$, and endo-map $f : C \to C$, there is a unique $h : \mathbb{N} \to C$ with $h(\mathsf{zero}) = c_0$ and $h(\mathsf{succ}\, n) = f(h\, n)$ for all $n : \mathbb{N}$.*

Proof. Existence: $h := \mathsf{rec}_{\mathbb{N}}\, C\, c_0\, (\lambda n.\, f)$. By the recursor's computation rules we have the *judgmental* equalities $h(\mathsf{zero}) \equiv c_0$ and $h(\mathsf{succ}\, n) \equiv f(h\, n)$.

Uniqueness: suppose $h' : \mathbb{N} \to C$ is given together with paths $p_0 : h'(\mathsf{zero}) = c_0$ and $p_s : \prod_{n:\mathbb{N}} h'(\mathsf{succ}\, n) = f(h'\, n)$ witnessing that h' also satisfies the equations (note these are *propositional* equalities – judgmental equality cannot be assumed for a generic h'). Define the predicate $P(n) := (h\,n =_C h'\,n)$.

Base case. Since $h(\mathsf{zero}) \equiv c_0$ judgmentally, the inverse path $p_0^{-1} : c_0 = h'(\mathsf{zero})$ inhabits $P(\mathsf{zero})$.

Inductive step. Assume $q : h\,n = h'\,n$. The recursor's computation rule gives $h(\mathsf{succ}\,n) \equiv f(h\,n)$ judgmentally, and applying $\mathsf{ap}_f : (a = b) \to (f\,a = f\,b)$ to q yields $\mathsf{ap}_f(q) : f(h\,n) = f(h'\,n)$. Composing with $p_s(n)^{-1}$ gives $f(h\,n) = h'(\mathsf{succ}\,n)$, hence $h(\mathsf{succ}\,n) = h'(\mathsf{succ}\,n)$. Apply $\mathsf{ind}_{\mathbb{N}}$ to conclude $\prod_{n:\mathbb{N}} P(n)$. Function extensionality (Theorem 5.7) then gives $h = h'$. □

3.3 $\mathbb{N}$ in HoTT vs. $\mathbb{N}$ in set theory

Remark 3.5. Proposition 3.4 renders Benacerraf's identification problem moot *within* the type theory: $\mathbb{N}$ is determined up to canonical bijection by its formation and elimination rules. Combined with univalence (Section 5) it is determined up to canonical *equality* as a term of $\mathcal{U}$.

4 Identity Types and Path Induction

4.1 The identity type

Definition 4.1 (Identity type). For each type $A : \mathcal{U}$ and elements $a, b : A$ there is a type $a =_A b$, called the *identity type* or *path space*. It is the inductive family generated by the single constructor $\mathsf{refl}_a : a =_A a$ for each $a : A$.

The eliminator – often called *path induction* or the *J-rule* – says that to prove a property of all paths it suffices to prove it of the constant paths.

Proposition 4.2 (Path induction, J). *For every type $A : \mathcal{U}$ and family $C : \prod_{x,y:A}(x =_A y) \to \mathcal{U}$, given a section $d : \prod_{x:A} C(x, x, \mathsf{refl}_x)$, there is a section*

$$J(C, d) : \prod_{x,y:A} \prod_{p:x=_A y} C(x, y, p),$$

satisfying the computation rule $J(C, d, x, x, \mathsf{refl}_x) \equiv d(x)$.

Remark 4.3. J is *not* the principle "every path is refl." That stronger statement is Streicher's K-axiom and is incompatible with univalence: it forces every type to be a set. HoTT keeps J but rejects K.

4.2 Symmetry, transitivity, and the path groupoid

Lemma 4.4 (Symmetry). *There is a function $\mathsf{sym} : \prod_{A:\mathcal{U}} \prod_{x,y:A}(x =_A y) \to (y =_A x)$.*

Proof. Take $C(x, y, p) := (y =_A x)$ and $d(x) := \mathsf{refl}_x$, then apply J. □

Lemma 4.5 (Transitivity)**.** *There is a function* $\mathsf{trans} : \prod_{A:\mathcal{U}} \prod_{x,y,z:A} (x =_A y) \to (y =_A z) \to (x =_A z)$*.*

Proof. Fix A, x, y, z and $p : x = y$. By J on p it suffices to assume $y \equiv x$ and $p \equiv \mathsf{refl}_x$, in which case $\mathsf{trans}(\mathsf{refl}_x, q) := q$ is a section. □

We write p^{-1} for $\mathsf{sym}(p)$ and $p \cdot q$ for $\mathsf{trans}(p, q)$.

Theorem 4.6 (Path groupoid)**.** *For any type* A *and* $a, b, c, d : A$*,* $p : a = b$*,* $q : b = c$*,* $r : c = d$*:*

1. $\mathsf{refl}_a \cdot p = p$ *and* $p \cdot \mathsf{refl}_b = p$*;*
2. $p \cdot p^{-1} = \mathsf{refl}_a$ *and* $p^{-1} \cdot p = \mathsf{refl}_b$*;*
3. $(p \cdot q) \cdot r = p \cdot (q \cdot r)$*.*

Moreover the higher coherences (the pentagon, the Eckmann–Hilton argument, etc.) all hold up to higher paths, making A *into an* ∞*-groupoid [3, 4].*

Proof sketch. Each clause is proved by J on the relevant paths. For the higher coherences one iterates J at successively higher levels; see [1, §2.1]. □

4.3 Transport

Definition 4.7 (Transport)**.** For $P : A \to \mathcal{U}$, $a, b : A$, and $p : a = b$, the function $\mathsf{transport}^P(p) : P(a) \to P(b)$ is defined by J: $\mathsf{transport}^P(\mathsf{refl}_a)(u) := u$.

Proposition 4.8 (Transport along concatenation)**.**

$$\mathsf{transport}^P(p \cdot q) = \mathsf{transport}^P(q) \circ \mathsf{transport}^P(p).$$

Proof. J on either p or q. □

4.4 Homotopy levels

Definition 4.9 (Contractibility, propositions, sets)**.** A type $A : \mathcal{U}$ is

- *contractible*, $\mathsf{isContr}(A) := \sum_{c:A} \prod_{x:A} (x = c)$;
- a *mere proposition*, $\mathsf{isProp}(A) := \prod_{x,y:A} (x = y)$;
- a *set* (h-set), $\mathsf{isSet}(A) := \prod_{x,y:A} \mathsf{isProp}(x = y)$.

The hierarchy continues: A is an n-type iff every identity type of A is an $(n-1)$-type.

Theorem 4.10 (Hedberg)**.** *If* A *has decidable equality* $\prod_{x,y:A} (x = y) + \neg(x = y)$*, then* A *is a set.*

Corollary 4.11. $\mathbb{N}$ *is a set.*

Proof. Equality on $\mathbb{N}$ is decidable by induction on both arguments and Peano's fourth axiom $\mathsf{succ}\, m = \mathsf{succ}\, n \Rightarrow m = n$. Apply Hedberg. □

5 The Univalence Axiom

5.1 Equivalence of types

Definition 5.1 (Quasi-inverse and equivalence)**.** A function $f : A \to B$ has a *quasi-inverse* if there exist $g : B \to A$ and homotopies $\eta : g \circ f \sim \mathrm{id}_A$ and $\varepsilon : f \circ g \sim \mathrm{id}_B$. The proposition $\mathsf{isEquiv}(f)$ is the half-adjoint version [1, §4.2], ensuring $\mathsf{isEquiv}(f)$ is itself a mere proposition. We write $A \simeq B := \sum_{f:A\to B} \mathsf{isEquiv}(f)$.

Proposition 5.2 (idtoeqv)**.** *There is a canonical map* $\mathsf{idtoeqv} : (A =_{\mathcal{U}} B) \to (A \simeq B)$ *taking* refl_A *to the identity equivalence.*

Proof. J on the path $A = B$. □

5.2 The axiom

Theorem 5.3 (Voevodsky's univalence axiom)**.** ([21]; [1, Axiom 2.10.3].) *For all types* $A, B : \mathcal{U}$*, the canonical map* $\mathsf{idtoeqv}$ *is itself an equivalence:*

$$\mathsf{ua} : (A \simeq B) \to (A =_{\mathcal{U}} B), \qquad \mathsf{idtoeqv} \circ \mathsf{ua} = \mathrm{id}_{A\simeq B}, \qquad \mathsf{ua} \circ \mathsf{idtoeqv} = \mathrm{id}_{A=_{\mathcal{U}}B}.$$

In Book HoTT this is taken as an axiom; in cubical type theory (Section 7) it is a theorem.

5.3 Consequences for arithmetic

Definition 5.4 (Type of NNOs)**.** Define

$$\mathrm{NNO} := \sum_{X:\mathcal{U}} \sum_{x_0:X} \sum_{s:X\to X} \mathsf{isInitial}(X, x_0, s),$$

where $\mathsf{isInitial}(X, x_0, s)$ is the universal property of Theorem 3.4 expressed as a Σ-type.

Theorem 5.5 (Contractibility of NNO)**.** NNO *is contractible.*

Proof. We exhibit $(\mathbb{N}, \mathsf{zero}, \mathsf{succ}, \mathsf{init}_{\mathbb{N}})$ as the centre and supply a proof that every other quadruple is equal to it.

Step 1. Equivalence at the carrier. By Theorem 3.4 the standard structure $(\mathbb{N}, \mathsf{zero}, \mathsf{succ})$ is initial. For any other initial (X, x_0, s), the universal property applied in both directions yields mutually inverse maps $f : \mathbb{N} \to X$ and $g : X \to \mathbb{N}$ with $g\circ f = \mathrm{id}_{\mathbb{N}}$ and $f\circ g = \mathrm{id}_X$ (each by uniqueness of the universal recursor). Half-adjointness packages this as an equivalence $e : \mathbb{N} \simeq X$. Univalence (Theorem 5.3) promotes this to a path $p := \mathsf{ua}(e) : \mathbb{N} =_{\mathcal{U}} X$.

Step 2. Paths in Σ-types. Recall that a path in $\sum_{a:A} B(a)$ between (a, b) and (a', b') is, up to canonical isomorphism, a pair (p, q) where $p : a =_A a'$ and $q : \mathsf{transport}^B(p)(b) =_{B(a')} b'$ [1, Theorem 2.7.2]. Iterating: a path between two iterated Σ-tuples $(a_1, a_2, \ldots, a_n)$ and $(a'_1, a'_2, \ldots, a'_n)$ is a tuple of paths $(p_1, p_2, \ldots, p_n)$ where p_k lives *over* the preceding $p_1, \ldots, p_{k-1}$.

Step 3. Constructing the path componentwise. We must produce paths (p, p_0, p_s, p_*) where:

1. $p : \mathbb{N} =_{\mathcal{U}} X$ is the carrier path from Step 1.

2. $p_0 : \mathsf{transport}^{(X \mapsto X)}(p)(\mathsf{zero}) =_X x_0$. Because $\mathsf{ua}(e)$ transports along the equivalence e, the left-hand side reduces to $e(\mathsf{zero})$, and the universal property of $\mathbb{N}$ forces $e(\mathsf{zero}) = x_0$ (the unique map $\mathbb{N} \to X$ sends zero to x_0), giving the required path.

3. $p_s : \mathsf{transport}^{(X \mapsto X \to X)}(p)(\mathsf{succ}) = s$. The type family $X \mapsto X \to X$ has transport given by conjugation by e, so the left-hand side reduces to $e \circ \mathsf{succ} \circ e^{-1}$. Naturality of the universal recursor yields $e \circ \mathsf{succ} = s \circ e$, whence $e \circ \mathsf{succ} \circ e^{-1} = s$, supplying p_s.

4. p_*: the initiality witness $\mathsf{isInitial}$ is a mere proposition (uniqueness of the universal recursor is a Pi-type into a contractible identity type), so any two initiality proofs are identified by a unique path.

Step 4. Conclusion. Bundling (p, p_0, p_s, p_*) via the iterated Σ-path constructor gives the required equality $(\mathbb{N}, \mathsf{zero}, \mathsf{succ}, \mathsf{init}_{\mathbb{N}}) =_{\mathrm{NNO}} (X, x_0, s, \cdot)$. Since $(X, x_0, s, \cdot)$ was arbitrary, NNO is contractible. The argument is the same shape as the contractibility of the type of "small structures-of-shape-T on a fixed base" once T is replaced by the signature "one constant and one unary operation"; see [1, §5.4 and §2.7]. □

Corollary 5.6. *For any NNO (X, x_0, s) the term $s^{58}\, x_0$ is, after transport along the canonical equality $\mathbb{N} = X$, equal to $\overline{58} : \mathbb{N}$.*

Theorem 5.6 is the formal expression of the slogan "58 is canonical across all NNOs."

5.4 Function extensionality

Theorem 5.7 (Function extensionality from univalence)**.** *If $f, g : \prod_{x:A} B(x)$ and there is a homotopy $h : \prod_{x:A}(f\,x = g\,x)$, then $f = g$.*

Proof sketch. Voevodsky proved this by representing $\prod_{x:A} B(x)$ via a clever Σ-type and applying univalence. See [1, §4.9]. □

6 Higher Inductive Types

A higher inductive type (HIT) extends inductive types with constructors for *paths* as well as points.

6.1 The interval and propositional truncation

Definition 6.1 (Interval)**.** The type I has constructors $0_I, 1_I : I$ and a path $\mathsf{seg} : 0_I = 1_I$.

Definition 6.2 (Propositional truncation)**.** $\|A\|$ has $|a| : \|A\|$ for $a : A$ and a path constructor $\prod_{x,y:\|A\|}(x = y)$. Its eliminator allows mapping $\|A\| \to P$ when P is a mere proposition.

6.2 The circle

Definition 6.3 (Circle). S^1 has constructors $\mathsf{base} : S^1$ and a path $\mathsf{loop} : \mathsf{base} =_{S^1} \mathsf{base}$. The eliminator: given $P : S^1 \to \mathcal{U}$, $b : P(\mathsf{base})$, and $\ell : \mathsf{transport}^P(\mathsf{loop})(b) = b$, there is $\mathsf{ind}_{S^1}(P, b, \ell) : \prod_{x:S^1} P(x)$, computing as $\mathsf{ind}_{S^1}(P, b, \ell)(\mathsf{base}) \equiv b$ and a corresponding rule on loop.

Theorem 6.4 (Loop space of S^1). *$\pi_1(S^1) := \|(\mathsf{base} = \mathsf{base})\|_0 \simeq \mathbb{Z}$. Indeed $(\mathsf{base} = \mathsf{base}) \simeq \mathbb{Z}$ as a type, before truncation.*

Proof sketch. Define the universal cover $\mathsf{code} : S^1 \to \mathcal{U}$ by $\mathsf{code}(\mathsf{base}) := \mathbb{Z}$ and on loop by univalence and the successor equivalence $\mathsf{succ} : \mathbb{Z} \simeq \mathbb{Z}$. Then $(\mathsf{base} = -) \sim \mathsf{code}$ by the encode–decode method (Licata–Shulman [5]); evaluating at base gives $(\mathsf{base} = \mathsf{base}) \simeq \mathbb{Z}$. □

Theorem 6.4 is the first non-trivial theorem of *synthetic homotopy theory*: a result of algebraic topology proved entirely inside type theory, without recourse to topological spaces, simplicial sets, or model categories.

6.3 Spheres, suspensions, and pushouts

The n-sphere S^n is defined recursively as ΣS^{n-1} where Σ is the unreduced suspension HIT. The pushout of $f : C \to A$ and $g : C \to B$ has constructors for A, B, and a path identifying $f(c)$ and $g(c)$ for each $c : C$.

Theorem 6.5 (Hopf fibration, internal). *There is a fibration $h : S^3 \to S^2$ with fibers S^1, defined internally as a type family $H : S^2 \to \mathcal{U}$ via the eliminator of the HIT S^2 (with total space $\sum_{x:S^2} H(x) \simeq S^3$), and characterized by the long exact sequence of homotopy groups; in particular $\pi_3(S^2) \simeq \mathbb{Z}$ via the Hopf invariant [1, §8.5].*

6.4 Eilenberg–MacLane spaces

For each abelian group G and $n \geq 1$ there is a HIT $K(G, n)$ with $\pi_n = G$ and $\pi_k = 0$ for $k \neq n$ [6]. Synthetic cohomology $H^n(X; G) := \|X \to K(G, n)\|_0$ recovers ordinary singular cohomology when X is a CW-complex.

6.5 Real numbers as a higher inductive–inductive type

Theorem 6.6 (Booij; HoTT Book §11.3). ([7], after [1, §11.3].) *There is a higher inductive–inductive type $\mathbb{R}_c$ presenting the* Cauchy reals*: an h-set with constructors for rational injections $\mathsf{rat} : \mathbb{Q} \to \mathbb{R}_c$, a Cauchy-completion limit operation, and a path constructor identifying terms whose distance witnesses are below every positive rational. $\mathbb{R}_c$ has the universal property of the Cauchy completion of $\mathbb{Q}$ in the category of metric spaces internal to HoTT, and satisfies the axioms of an Archimedean ordered field.*

The construction is *higher* because of the path constructor and *inductive–inductive* because the path constructor mentions a "closeness" relation defined simultaneously with the carrier type.

6.6 Worked example: the canonical 58 across NNOs

To make Theorem 5.5 concrete, we walk through how an arbitrary NNO yields the same "58" as $\mathbb{N}$.

Example 6.7 (58 in two NNOs)**.** Take two structures.

- The standard one: $(\mathbb{N}, \mathsf{zero}, \mathsf{succ})$.
- A bit-string encoding: $X := \mathrm{List}(\mathbf{1})$, with $x_0 := \mathsf{nil}$ and $s := \mathsf{cons}(\star, -)$.

Both are easily checked to satisfy the universal property of Theorem 3.4; for the bit-string version, the unique recursor sends a list of length n to the value $f^n(c_0)$ in any target (C, c_0, f).

By Theorem 5.5, the data $(\mathbb{N}, \mathsf{zero}, \mathsf{succ})$ and (X, x_0, s) are equal as inhabitants of NNO. Concretely:

1. the equivalence $e : \mathbb{N} \simeq X$ sends $\overline{n}$ to the list of length n;
2. univalence gives $\mathsf{ua}(e) : \mathbb{N} =_{\mathcal{U}} X$;
3. transporting $\overline{58} : \mathbb{N}$ along $\mathsf{ua}(e)$ yields the 58-element list of $\star$'s in X;
4. this list is, by definition of X, the value of $s^{58}\, x_0$.

Thus the two "58"s are propositionally equal, and *this very equality* is the content of Theorem 5.6.

Remark 6.8 (The Yoneda perspective recovered)*.* Paper 04 characterises every object A by its representable functor $\mathrm{Hom}(-, A)$ and shows $A \simeq B \Leftrightarrow \mathrm{Hom}(-, A) \simeq \mathrm{Hom}(-, B)$ (Yoneda). Univalence upgrades the right-to-left direction: *equivalent representables yield equal objects in $\mathcal{U}$*, which is the type-theoretic statement of the structuralist slogan "a thing is what it relates to."

6.7 Worked example: the loop concatenation as integer addition

For Theorem 6.4, let us trace what happens to specific loops.

Example 6.9 (Three full turns then two backward)**.** Inside HoTT we represent the loop "three times around" as the path $\mathsf{loop} \cdot \mathsf{loop} \cdot \mathsf{loop}$ at base in S^1, i.e. a term of $\mathsf{base} = \mathsf{base}$. By the encode–decode equivalence $(\mathsf{base} = \mathsf{base}) \simeq \mathbb{Z}$, this term encodes to the integer 3.

Composing with $\mathsf{loop}^{-1} \cdot \mathsf{loop}^{-1}$ gives, by Theorem 4.6, the path $\mathsf{loop} \cdot \mathsf{loop} \cdot \mathsf{loop} \cdot \mathsf{loop}^{-1} \cdot \mathsf{loop}^{-1}$, which the encode-decode equivalence sends to $3 + (-1) + (-1) = 1$.

Decoding back yields a single loop, propositionally equal to the original concatenation. The point is not that we computed $3 - 2 = 1$ (routine!), but that this very arithmetic of $\mathbb{Z}$ *is* the fundamental group law on $\pi_1(S^1)$ once we identify the two via the equivalence.

7 Cubical Type Theory

The original presentation of HoTT takes univalence as an axiom, breaking canonicity: a closed term of $\mathbb{N}$ may be propositionally but not definitionally equal to a numeral. Cubical type theory restores canonicity.

7.1 Intervals and faces

Definition 7.1 (Cube category)**.** Cohen–Coquand–Huber–Mörtberg [8] introduce a primitive interval object $\mathbb{I}$ generated by $0, 1 : \mathbb{I}$, the operations $\sqcup, \sqcap, \neg$ making $\mathbb{I}$ into a de Morgan algebra, and the algebra of *face formulas* φ over the generating equations $i = 0$, $i = 1$.

A path $p : a =_A b$ is then a function $\mathbb{I} \to A$ with $p(0) = a$ and $p(1) = b$ judgmentally. Path concatenation, inversion, and dependent elimination become explicit term operations on cubical sets, not derived from J.

7.2 Composition and Glue

The crucial operations are *composition* (filling open boxes) and *Glue* (gluing types along an equivalence). The Glue type $\mathsf{Glue}\, A\, [\varphi \mapsto (T, f)]$ where $f : T \simeq A$ on the face φ supplies a constructive proof of univalence: every equivalence is realized as a path in $\mathcal{U}$.

Theorem 7.2 (Univalence in cubical type theory)**.** *Cubical type theory in the CCHM model proves* ua *as a defined term, and* $\mathsf{idtoeqv} \circ \mathsf{ua}$ *reduces to the identity by computation, not by postulate [8, 9].*

7.3 Implementations

The Cubical Agda mode [9] and the `cubicaltt` prototype implement the system; the `redtt` and `cooltt` projects extend it with cartesian intervals and modalities. For our purposes the relevant fact is that all theorems of Sections 5 and 6 can be *computed*, not just postulated.

8 Variants: 2LTT, Modal, and Directed HoTT

8.1 Two-level type theory

Two-level type theory (Annenkov–Capriotti–Kraus–Sattler [14]) introduces a strict equality alongside the homotopical one. The strict fragment is used to formalize semantics (e.g. inverse diagrams of types representing semisimplicial types) that the homotopical fragment alone cannot encode without higher coherence. Strict equality is incompatible with univalence, so 2LTT carefully separates the two universes.

8.2 Modal HoTT

Modal type theory adds operators $\Box, \Diamond$ corresponding to adjunctions in the categorical semantics. Examples include Schreiber–Shulman cohesive HoTT [15], where the cohesion modalities recover the discrete–codiscrete–shape adjoint string $\mathrm{disc} \dashv \Pi \dashv \mathrm{codisc}$ for cohesive $(\infty, 1)$-toposes, and the spatial type theory of Riley [16].

8.3 Directed HoTT

Directed type theory replaces invertible paths with possibly non-invertible *morphisms*, so that types correspond to (∞, n)-categories rather than ∞-groupoids. Riehl–Shulman [17] develop *simplicial type theory* with a primitive directed interval Δ^1 and prove a synthetic Yoneda lemma; directed univalence [18] is an active research frontier.

9 Transcendentals in HoTT

We work inside $\mathbb{R}_c$ from Section 6.5.

9.1 π as the period of $\exp$ on $i\mathbb{R}$

Definition 9.1 (Real exponential)**.** The map $\exp : \mathbb{R}_c \to \mathbb{R}_c$ is defined as the unique solution to $y' = y, y(0) = 1$ in $\mathbb{R}_c$ – equivalently, by the power series $\sum_{n=0}^{\infty} x^n/n!$, which converges in $\mathbb{R}_c$ by completeness.

Definition 9.2 (Pi)**.** Define $\sin : \mathbb{R}_c \to \mathbb{R}_c$ by the power series $\sum_{k=0}^{\infty}(-1)^k x^{2k+1}/(2k+1)!$. The unique-existence proposition

$$\sum_{p:\mathbb{R}_c} \big(\sin(p) = 0\big) \times \big(p > 0\big) \times \Big(\prod_{x:\mathbb{R}_c} 0 < x < p \to \sin(x) > 0\Big)$$

is contractible. We define π to be its centre.

Theorem 9.3. *Lindemann's theorem ports to HoTT: π is transcendental over $\mathbb{Q}$, in the sense that the type*

$$\sum_{p:\mathrm{Poly}(\mathbb{Q})} \big(p \neq 0\big) \times \big(p(\pi) = 0\big)$$

is empty (equivalent to $\mathbf{0}$*).*

Proof sketch. Lindemann's classical proof relies only on convergent power series, elementary symmetric functions, and bounding arguments – all of which are available constructively in $\mathbb{R}_c$. The port is therefore believed to be possible based on the constructive character of the classical argument; however, to the authors' knowledge no end-to-end formalization in HoTT is yet complete. The presently available scaffolding includes:

- Cauchy completeness of $\mathbb{R}_c$, basic limits, the Archimedean property, and elementary calculus on $\mathbb{R}_c$ (Booij's thesis [7], expanding [1, §11.3]).
- Convergence of power series and analytic functions in HoTT, plus algebraic manipulations of polynomial rings and symmetric functions (scaffolded in the agda-unimath project [19]).

The pieces still missing for an end-to-end formalization are: (i) the Hermite–Lindemann auxiliary function $J(t) = \int_0^t e^{t-s} f(s)\, ds$ and its bounds in $\mathbb{R}_c$; (ii) the algebraic-integer estimates that drive the contradiction; and (iii) the integration with a finished formalization of the Cauchy integral over $\mathbb{R}_c$. We identify this as open work in Section 15. $\square$

9.2 e as the unique fixed point of D

Definition 9.4 (Euler's number). Let $\exp : \mathbb{R}_c \to \mathbb{R}_c$ be as above. Then $e := \exp(1)$. Equivalently, e is the centre of the contractible type $\sum_{x:\mathbb{R}_c} \sum_{f:\mathbb{R}_c \to \mathbb{R}_c} (f' = f) \times (f(0) = 1) \times (x = f(1))$.

Theorem 9.5 (Hermite, ported). *e is transcendental over $\mathbb{Q}$.*

9.3 Identification across encodings

Theorem 9.6 (Reals are unique in HoTT up to equivalence). *Any two h-sets satisfying the universal property of the Cauchy reals (or the Dedekind reals, modulo the law of excluded middle for the latter) are equivalent, hence equal as types by univalence. Consequently, π and e are well-defined independently of the chosen construction.*

10 Bridge: HoTT as the Internal Language of $(\infty, 1)$-Toposes

10.1 Categorical semantics

Theorem 10.1 (Awodey–Warren). ([10].) *Identity types of intensional Martin-Löf type theory are interpreted in weak factorisation systems; the model in simplicial sets exhibits the path groupoid as a Kan complex.*

Voevodsky's simplicial set model [11] establishes that HoTT with univalence has a sound interpretation in the $(\infty, 1)$-topos sSet of Kan complexes.

Theorem 10.2 (Shulman). ([12].) *Every $(\infty, 1)$-topos $\mathcal{E}$ admits a model of HoTT with univalence; conversely, the syntactic category of HoTT is conjecturally the initial $(\infty, 1)$-topos.*

10.2 Co-extensiveness with categorical and structural perspectives

The Yoneda embedding (Paper 04) and the categorical structuralism of Paper 06 have always been "equivalent up to equivalence." Univalence makes them *equal* – equal as objects of $\mathcal{U}$, equal as theories under semantic interpretation, equal as derivations modulo a translation that is itself an equivalence.

Remark 10.3 (The univalent correspondence, formalized). The Curry–Howard–Lambek tradition (see Wadler's exposition [22]) identifies logic, type theory, and category theory as a triple. The four-column correspondence

$$\text{Logic} \longleftrightarrow \text{Type Theory} \longleftrightarrow \text{Category Theory} \longleftrightarrow \text{Homotopy}$$

is, in HoTT, the statement that the four interpretations of the same term in $\mathcal{U}$ correspond to the same homotopy type, with the categorical column interpreted in $(\infty, 1)$-toposes (Lurie's higher topos theory [23] provides the ambient setting). In particular, "58 as a proposition" (the inhabited type $\mathsf{Fin}_{59} \cap \mathsf{prime}^c$, etc.), "58 as a term" ($\overline{58} : \mathbb{N}$), "58 as an object" (the discrete object $58 \in \mathsf{Set}$), and "58 as a space" (the discrete homotopy type $\mathsf{B1}^{58}$) are equal up to canonical paths.

11 Results

We collect the principal contributions of this paper.

Theorem 11.1 (Encoding-free arithmetic)**.** *In Book HoTT with univalence, the following are equivalent:*

1. *the inductive type* $\mathbb{N}$ *with* $\mathsf{zero}, \mathsf{succ}$*;*
2. *the initial NNO in any model of HoTT with* $\mathbb{N}$*-types;*
3. *the unique pointed endo-structure* X *with* $\mathsf{isInitial}(X, x_0, s)$*;*
4. *up to univalent equality, every other type satisfying any of (1)–(3).*

The witnesses to these equivalences are themselves canonical (paths in $\mathcal{U}$*), so* $58 := \overline{58}$ *is independent of representation.*

Theorem 11.2 (Synthetic computation of $\pi_1(S^1)$)**.** *Within HoTT,* $\pi_1(S^1) = \mathbb{Z}$ *admits a fully formal proof [5] that mechanizes in Agda and Cubical Agda; the proof uses no external topology, only the HIT* S^1 *and univalence.*

Theorem 11.3 (Reals and transcendentals)**.** *The Cauchy reals* $\mathbb{R}_c$ *form an h-set inside HoTT (Theorem 6.6). The terms* π *and* e *are unique-existence centres of contractible types of solutions of structural conditions. Their classical properties (irrationality, transcendence) port mechanically.*

Theorem 11.4 (Bridge theorem)**.** *HoTT is sound and complete for* $(\infty, 1)$*-topos semantics [12]. Hence the categorical (Paper 06), structural (Paper 07), Yoneda (Paper 04), and HoTT (Paper 05) views of arithmetic are co-extensive: a statement provable in any one is provable in any other, modulo translation.*

12 Discussion

12.1 What HoTT does not solve

HoTT is a foundation, not a panacea. The following remain open:

- *Coherence beyond truncation level*: proving general coherence theorems for $(\infty, 1)$-categories *inside* HoTT (rather than via 2LTT) is non-trivial. The semisimplicial types problem [20] is the canonical example.

- *Decidability and computation under axioms*: Book HoTT loses canonicity. Cubical type theory restores it but at the cost of a more intricate calculus.

- *Set-theoretic strength*: HoTT with univalence has the proof-theoretic strength of ZFC + "there exist many inaccessible cardinals" [13], comparable to standard mathematical practice. Whether all of analytic number theory ports remains an open research front (Paper 07 surveys this).

12.2 Philosophical reading

The shift from set theory to HoTT mirrors the shift from substantialism to structuralism. "58 is a thing" becomes "58 is a position in a structure" becomes "58 is the equivalence class of all positions in isomorphic structures" becomes (in HoTT) "58 is a term whose identity is witnessed by paths." Univalence collapses the apparent distinction between the last two.

12.3 Pedagogical implications

Synthetic homotopy theory provides a path-from-paths approach to algebraic topology: $\pi_1(S^1) = \mathbb{Z}$ in twelve lines of Agda. This is not merely a streamlining; it indicates that homotopy is more primitive than topology in the sense that the latter can be dispensed with.

12.4 Computational realisation

The Haskell companion (Section 13) and the Agda fragments (Section 14) demonstrate that path induction, transport, and the basic equational reasoning calculus can be *simulated* at the type level in mainstream functional languages, even without full dependent types. This is a useful pedagogical bridge, not a replacement.

12.5 Comparison with set-theoretic and categorical foundations

It is illuminating to put the four foundational stances side by side, as they answer the question "what is the natural number 58?".

- **Set theory (Paper 02).** "58 is the von Neumann ordinal $\{0, 1, \ldots, 57\}$." The encoding is canonical only by stipulation; Benacerraf's identification problem records the residual arbitrariness.

- **Categorical foundations (Paper 03 / Paper 04).** "58 is the value of the unique recursion morphism on the natural-number object, equivalently a position in the representable $\mathrm{Hom}(-, \mathbb{N})$." Encoding-independence holds up to canonical isomorphism; the question "but *which* NNO?" remains a residue, albeit a small one.

- **HoTT (this paper).** "58 is $\overline{58} : \mathbb{N}$, where $\mathbb{N}$ is determined up to a canonical *equality* of types by univalence." Encoding-independence is internal: there is no extra- theoretic notion of "which $\mathbb{N}$."

- **Structural / ∞-topos perspective (Paper 06).** "58 is an invariant under all structure-preserving maps," which by the internal-language theorem is the same as the HoTT statement.

The progression is one of *decreasing residue*: each step removes a notion that the previous level could not eliminate.

12.6 When axiomatic univalence cannot be avoided

In Book HoTT, univalence is *not* a constructive principle: a closed term of $\mathbb{N}$ produced via univalence may fail to reduce to a numeral. For many proofs this matters little (we only care about propositional equality), but for executable extraction it is fatal. Cubical type theory solves this; some applications – notably synthetic differential geometry [15] – additionally require modal axioms that themselves are not yet known to be cubically computational. The trade-off between axiomatic generality and definitional reduction is an active research front.

13 Companion Implementation

The companion Haskell library is available in the project's public source repository (under `src/05-hott/`; see the *Univalent Correspondence* project). It provides:

- A GADT `Path a b` representing identity types with constructor `Refl :: Path a a`.
- Functions `sym`, `trans`, and `transport` mirroring Theorems 4.4, 4.5 and 4.7.
- A type-level naturals module with peano-style `succ` and a demonstration that 58 can be constructed and equal-typed.
- A module sketching the circle as a HIT (necessarily approximated, as Haskell lacks higher constructors) and the encode–decode method.
- An Agda-style sketch in comments showing the analogous direct type-theoretic syntax.

The `Main.hs` executable runs a small demonstration.

14 Agda Sketch

The most natural ambient language for HoTT is Agda or Cubical Agda. We include here an Agda-flavoured sketch of the basic constructions; full files for Cubical Agda live alongside the Haskell sources.

```
-- Agda-style (comments only; the actual Agda lives elsewhere):
--
-- data Nat : Set where
-- zero : Nat
-- succ : Nat -> Nat
--
-- _==_ : {A : Set} -> A -> A -> Set
```

```
-- a == b = Path a b
--
-- refl : {A : Set} {a : A} -> a == a
-- refl = Refl
--
-- sym : {A : Set} {a b : A} -> a == b -> b == a
-- sym Refl = Refl
--
-- trans : {A : Set} {a b c : A}
-- -> a == b -> b == c -> a == c
-- trans Refl q = q
--
-- transport : {A : Set} (P : A -> Set)
-- -> {a b : A} -> a == b -> P a -> P b
-- transport P Refl x = x
--
-- ua : {A B : Set} -> A ~= B -> A == B
-- (postulated in Book HoTT, derived in Cubical)
```

15 Conclusion and Future Work

We have argued that Homotopy Type Theory provides the synthetic language in which arithmetic, analysis, and algebraic topology share a single foundational system. The natural number 58 is the canonical $\overline{58} : \mathbb{N}$; the universe-level identification of all NNOs is a theorem; the transcendental constants π and e are unique-existence centres in the higher inductive–inductive type $\mathbb{R}_c$. Univalence makes the categorical and structural perspectives co-extensive with the type-theoretic one.

The companion Haskell code provides a hands-on, executable mirror of the key constructions. The next paper (Paper 06) develops the categorical / structural perspective independently and shows that it merges with the HoTT perspective via the internal language theorem; Paper 07 (the synthesis) closes the loop.

15.1 Open problems

1. Formalize Lindemann–Weierstrass in Cubical Agda end-to-end.
2. Develop directed univalence sufficient to express the synthetic $(\infty, 2)$-Yoneda lemma.
3. Formalize the Brown representability theorem inside synthetic homotopy theory; this would clarify the relation to the spectra constructions in classical algebraic topology.

References

[1] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study, 2013. `arXiv:1308.0729`.

[2] P. Martin-Löf. *Intuitionistic Type Theory*. Bibliopolis, 1984.

[3] P. L. Lumsdaine. Weak omega-categories from intensional type theory. *Logical Methods in Computer Science*, 6(3:24):1–19, 2010.

[4] B. van den Berg and R. Garner. Types are weak omega-groupoids. *Proc. London Math. Soc.*, 102(2):370–394, 2011.

[5] D. R. Licata and M. Shulman. Calculating the fundamental group of the circle in homotopy type theory. In *LICS 2013*, pp. 223–232.

[6] D. R. Licata and E. Finster. Eilenberg–MacLane spaces in homotopy type theory. In *CSL-LICS 2014*, ACM, 2014.

[7] A. B. Booij. *Analysis in univalent type theory.* PhD thesis, University of Birmingham, 2020.

[8] C. Cohen, T. Coquand, S. Huber, A. Mörtberg. Cubical type theory: a constructive interpretation of the univalence axiom. In *TYPES 2015*, LIPIcs 69, pp. 5:1–5:34.

[9] A. Vezzosi, A. Mörtberg, A. Abel. Cubical Agda: a dependently typed programming language with univalence and higher inductive types. *Proc. ACM Program. Lang.*, 3(ICFP):87:1–87:29, 2019.

[10] S. Awodey and M. A. Warren. Homotopy theoretic models of identity types. *Math. Proc. Cambridge Philos. Soc.*, 146(1):45–55, 2009.

[11] C. Kapulkin and P. L. Lumsdaine. The simplicial model of univalent foundations (after Voevodsky). *J. Eur. Math. Soc.*, 23(6):2071–2126, 2021.

[12] M. Shulman. All $(\infty, 1)$-toposes have strict univalent universes. `arXiv:1904.07004`, 2019.

[13] M. Shulman. Higher inductive types and the formal Blakers–Massey theorem. *Proc. ACM Program. Lang.*, 4(POPL):41:1–41:30, 2020.

[14] D. Annenkov, P. Capriotti, N. Kraus, C. Sattler. Two-level type theory and applications. *Math. Structures Comput. Sci.*, 33(8):688–743, 2023.

[15] U. Schreiber and M. Shulman. Quantum gauge field theory in cohesive homotopy type theory. In *QPL 2012*, EPTCS 158, pp. 109–126, 2014.

[16] M. Riley. *A type theory with a tiny object.* PhD thesis, Wesleyan University, 2022.

[17] E. Riehl and M. Shulman. A type theory for synthetic ∞-categories. *Higher Structures*, 1(1):147–224, 2017.

[18] D.-C. Cisinski, H. K. Nguyen. The universal coCartesian fibration. `arXiv:2210.08945`, 2022.

[19] The agda-unimath project. *The agda-unimath library.* `https://unimath.github.io/agda-unimath/`, accessed 2026.

[20] N. Kraus. The general universal property of the propositional truncation. *TYPES 2014*, LIPIcs 39, pp. 111–145.

[21] V. Voevodsky. The equivalence axiom and univalent models of type theory. `arXiv:1402.5556`, 2014.

[22] P. Wadler. Propositions as types. *Comm. ACM*, 58(12):75–84, 2015.

[23] J. Lurie. *Higher Topos Theory*. Annals of Math. Studies 170, Princeton, 2009.