

Univalence without function extensionality^{*}

Evan Cavallo^{a,1,3} Jonas Höfer^{a,2}

^a *Department of Computer Science and Engineering
University of Gothenburg and Chalmers University of Technology
Gothenburg, Sweden*

Abstract

It is a well-known theorem of homotopy type theory, originally due to Voevodsky, that function extensionality holds inside any univalent universe. We consider a weaker variant of the univalence axiom, asserting that the wild category formed by the universe is univalent, which we call categorical univalence. We show that categorical univalence does not imply function extensionality by an analysis of Von Glehn’s polynomial model construction, which produces models of Martin-Löf type theory that always refute function extensionality. We find in particular that when the base model has a univalent universe, its polynomial model has a universe that is categorically univalent but lacks function extensionality.

Keywords: univalence, function extensionality, homotopy type theory, type theory, polynomial functor

1 Introduction

In 2010, Voevodsky [47,48] discovered that any universe of intensional Martin-Löf type theory (ITT) satisfying his univalence axiom also satisfies function extensionality: (dependent) functions between types in the universe are equal as soon as they are homotopic. This result became a foundational pillar of Homotopy Type Theory / Univalent Foundations. For constructivists, it was an additional motivation to justify univalence constructively—noted for example by Bezem, Coquand, and Huber [7]—given the historical difficulty of integrating function extensionality with constructive type theory.

At the same time, the connection between univalence and function extensionality has always seemed contingent. It is unclear whether univalence implies extensionality principles for other negative type formers, such as coinductive [49] or modal [24, Conjecture 11.2.2] types, which suggests functions might be privileged simply because they appear in the statement of univalence. Furthermore, minor variations on the univalence axiom are not known to imply function extensionality.

In a post on MathOverflow in 2013 [18], François G. Dorais proposed⁴ one such variation. To contextualize Dorais’ axiom, let us first review the standard definitions. For functions $f, g: \prod_{a:A} B(a)$, we write $f \sim g := \prod_{a:A} fa =_{B(a)} ga$ for the type of *homotopies* from f to g . The type $A \simeq B$ of (*homotopy*)

^{*} We thank Lorenzo Perticone for first (inadvertently) calling our attention to this question, and we thank the Gothenburg Logic and Types unit for many lunchtime discussions on the topic. We also thank András Kovács for his Agda formalization of the polynomial model, which was of great help to us in understanding the construction.

¹ Email: evan.cavallo@gu.se

² Email: hoferj@chalmers.se

³ Supported by the Knut and Alice Wallenberg Foundation (KAW), Grant No. 2019.0116

⁴ With some input from Mike Shulman.

equivalences between types A, B is the type of homotopy bi-invertible maps, that is, $f: A \rightarrow B$ equipped with $s, r: B \rightarrow A$ such that $fs \sim \text{id}_B$ and $rf \sim \text{id}_A$ [36, §9.2]. We assume a fixed universe \mathcal{U} .

Definition 1.1 *Function extensionality* (FE) is the principle that for every family of types $a: A \vdash B(a)$ and $f, g: \prod_{a:A} B(a)$, the map $(f =_{\prod_{a:A} B(a)} g) \rightarrow (f \sim g)$ is an equivalence. We write $\text{FE}_{\mathcal{U}}$ for the relativization of FE to \mathcal{U} , i.e., its restriction to the case where $A: \mathcal{U}$ and $B: A \rightarrow \mathcal{U}$.

Definition 1.2 *Univalence* ($\text{UA}_{\mathcal{U}}$) is the principle that the map $\text{id-to-eq}: (A =_{\mathcal{U}} B) \rightarrow (A \simeq B)$ is an equivalence for all $A, B: \mathcal{U}$.

Dorais observed essentially that the map $\text{id-to-eq}: (A =_{\mathcal{U}} B) \rightarrow (A \simeq B)$, which sends the reflexive path to the identity equivalence, factors up to homotopy through an intermediate type

$$(A =_{\mathcal{U}} B) \begin{array}{c} \xrightarrow{\text{id-to-ceq}} (A \cong B) \xrightarrow{\text{ceq-to-eq}} \\ \xrightarrow{\text{id-to-eq}} (A \simeq B) \end{array}$$

of what we call *categorical equivalences*: maps $f: A \rightarrow B$ equipped with $s, r: B \rightarrow A$ such that $fs =_{B \rightarrow B} \text{id}_B$ and $rf =_{A \rightarrow A} \text{id}_A$, i.e., with left and right inverses up to equality rather than homotopy. This suggests Dorais' proposed weakening of univalence:

Definition 1.3 *Categorical univalence* ($\text{CUA}_{\mathcal{U}}$) is the principle that $\text{id-to-ceq}: (A =_{\mathcal{U}} B) \rightarrow (A \cong B)$ is an equivalence for all $A, B: \mathcal{U}$.

The type $A \cong B$ can be described as the type of isomorphisms from A to B in the *wild category* of types in \mathcal{U} and functions between them. $\text{CUA}_{\mathcal{U}}$ states exactly that \mathcal{U} is a univalent wild category. In the presence of function extensionality in \mathcal{U} , the map $(A \cong B) \rightarrow (A \simeq B)$ is an equivalence, and so $\text{FE}_{\mathcal{U}} + \text{CUA}_{\mathcal{U}}$ implies $\text{UA}_{\mathcal{U}}$; conversely, the fact that $\text{UA}_{\mathcal{U}}$ implies $\text{FE}_{\mathcal{U}}$ means that it also implies $\text{CUA}_{\mathcal{U}}$. Dorais asked whether the converse is true: does $\text{CUA}_{\mathcal{U}}$ imply $\text{UA}_{\mathcal{U}}$, or equivalently $\text{FE}_{\mathcal{U}}$?

We answer this question in the negative, identifying a model of ITT with a universe that validates $\text{CUA}_{\mathcal{U}}$ but not $\text{FE}_{\mathcal{U}}$. Actually, we prove the consistency of $\neg \text{FE}_{\mathcal{U}}$ with a slightly stronger statement:

Definition 1.4 *Familial categorical univalence* ($\text{CUA}_{\mathcal{U}}^{\bullet}$) is the principle that for all $I: \mathcal{U}$, the wild category \mathcal{U}^I —whose objects are families $A: I \rightarrow \mathcal{U}$ and whose morphisms $A \rightarrow B$ are families of functions $\prod_{i:I} A(i) \rightarrow B(i)$ —is a univalent wild category.

We assume strict η laws for unit and Π types, so $\text{CUA}_{\mathcal{U}}^{\bullet}$ implies $\text{CUA}_{\mathcal{U}}$ by taking $I = 1$. We show the independence of $\text{CUA}_{\mathcal{U}}^{\bullet}$ from $\text{FE}_{\mathcal{U}}$ using Von Glehn's *polynomial model* construction $\mathbf{Poly}(-)$ [50, 33], a known source of models of type theory that refute function extensionality. Specifically, we prove:

Theorem 1.5 (4.17) *Let \mathbb{C} be a model of ITT with extensive finite coproducts of types satisfying the strict η rule. If $\mathbb{C} \models \text{CUA}_{\mathcal{U}}^{\bullet}$, then $\mathbf{Poly}(\mathbb{C}) \models \text{CUA}_{\mathcal{U}}^{\bullet}$.*

Familial categorical univalence arises naturally in the construction: just to show $\mathbf{Poly}(\mathbb{C}) \models \text{CUA}_{\mathcal{U}}$, we already require $\mathbb{C} \models \text{CUA}_{\mathcal{U}}^{\bullet}$. Function extensionality always fails in polynomial models [50, §4.5], so it remains to provide a suitable input model. Off-the-shelf cubical and simplicial models of homotopy type theory will do, as Moss and Von Glehn have already observed [33, §6]. We conclude:

Theorem 1.6 (5.6) $\text{ITT} + \text{CUA}_{\mathcal{U}}^{\bullet} \not\models \text{FE}_{\mathcal{U}}$.

Part of the appeal of weak foundations is that they allow us to tease apart the components of mathematics. Each type former of Martin-Löf's type theory has a distinct, well-defined purpose. Univalence fits in uneasily in this picture. While it has beautiful consequences, it also has *many* consequences, and—like impredicativity or the law of the excluded middle—it may be hiding finer structure beneath its surface.

By scratching at that surface, we hope to understand what makes univalence tick. The polynomial model offers some motivation and a testing ground for weaker forms of the axiom. We are left with more questions than answers; unlike in the case with FE, where superficial variations on univalence usually turn out to be equivalent, here we find subtly distinct axioms with no canonical choice among them. Still, we hope our results can provoke further reflection on the foundations of homotopy type theory.

1.1 Outline

In Section 2, we recall some basic definitions, then observe that $\text{FE}_{\mathcal{U}}$ holds if and only if the canonical map $\text{ceq-to-eq}: (A \cong B) \rightarrow (A \simeq B)$ is an equivalence for all $A, B: \mathcal{U}$, meaning that univalence quite literally factors into function extensionality and categorical univalence. In Section 3 we recall Von Glehn’s polynomial model construction. The main technical contribution is Section 4, where we show that the polynomial model $\mathbf{Poly}(\mathbb{C})$ inherits $\text{CUA}_{\mathcal{U}}^{\bullet}$ from the base model \mathbb{C} . In Section 5 we apply this result to a univalent base model to conclude that $\text{ITT} + \text{CUA}_{\mathcal{U}}^{\bullet} \not\vdash \text{FE}_{\mathcal{U}}$. We discuss and compare other possible weakenings of $\text{UA}_{\mathcal{U}}$ in Section 6, and finish with a review of related work in Section 7.

2 Decomposing univalence

Our basic theory ITT is Martin-Löf type theory with Σ types, Π types, intensional identity types, binary coproduct types, empty and unit types, and one universe \mathcal{U} closed under all of these type formers.⁵ We use the term *strict equality* and symbol $\overset{\circ}{=}$ for equality on the judgmental level; we use $=$ for identity types. We use $\overset{\circ}{\cong}$ for strict isomorphisms: two functions in opposite directions composing strictly to the identity. Note that $A \overset{\circ}{\cong} B$ is not a type, and $e: A \overset{\circ}{\cong} B$ is merely a shorthand for a meta-level assumption. Besides strict β rules for all type formers, we include strict η rules for Σ types, Π types, and the unit type.

For basic results, we cite Rijke’s book [36], which does not introduce FE until Chapter 13; we only use results from earlier chapters. Crucially, we have basic facts about contractible types and that Σ types respect equivalences in both arguments. Note that the analogous fact does not hold for Π types absent FE. In contrast to Rijke [36], we assume the strict η rule for Σ types, not only for Π types. This means, for example, that the equivalence witnessing the distributivity of Π types over Σ types is a strict isomorphism.

2.1 Univalent wild categories

The universe of ITT has the structure of an $(\infty, 1)$ -category, with type of objects $\mathcal{U}_0 := \mathcal{U}$ and type of morphisms $\mathcal{U}_1(A, B) := (A \rightarrow B)$. The first layer of such an $(\infty, 1)$ -categorical structure is captured by the Capriotti and Kraus’ notion of *wild category* [11, Definition 4.1].

Definition 2.1 A *wild category*⁶ \mathbb{C} is a type \mathbb{C}_0 and family of types $x, y: \mathbb{C}_0 \vdash \mathbb{C}_1(x, y)$ equipped with

- (i) composites $g \circ f: \mathbb{C}_1(x, z)$ for all $g: \mathbb{C}_1(y, z), f: \mathbb{C}_1(x, y)$,
- (ii) identities $\text{id}_x: \mathbb{C}_1(x, x)$ for all $x: \mathbb{C}_0$,
- (iii) associators $\alpha_{h,g,f}: h \circ (g \circ f) = (h \circ g) \circ f$ for all $h: \mathbb{C}_1(z, w), g: \mathbb{C}_1(y, z), f: \mathbb{C}_1(x, y)$, and
- (iv) unitors $\lambda_f: \text{id}_y \circ f = f$ and $\rho_f: f \circ \text{id}_x = f$ for all $f: \mathbb{C}_1(x, y)$.

If clear from context, we omit the subscripts when referring to the type of objects or family of morphisms. We write $x \rightarrow y$ for $\mathbb{C}_1(x, y)$ when \mathbb{C} is clear, and we sometimes write gf for $g \circ f$.

Example 2.2 As noted above, the universe \mathcal{U} has a wild category structure with $\mathcal{U}(A, B) := (A \rightarrow B)$, composition and identities given by the usual composition of functions and identity functions, and reflexive equalities for the associators and unitors. More generally, for every type I there is a wild category \mathcal{U}^I whose objects are families $A: I \rightarrow \mathcal{U}$ and whose morphisms are indexed functions, $\mathcal{U}^I(A, B) := \prod_{i:I} A(i) \rightarrow B(i)$.

These wild categories are really strictly coherent $(\infty, 1)$ -categories: the associators and unitors are strict equalities and satisfy all higher coherence laws (e.g., the pentagon) up to strict equality. All of the concrete wild categories we encounter in this article are of this kind.

Importantly, wild-categorical structure suffices to define *isomorphism*.

Definition 2.3 Given $s: x \rightarrow y$ and $r: y \rightarrow x$ in a wild category \mathbb{C} , we say that r is a *retraction* of s and s is a *section* of r if $rs = \text{id}_x$. For a morphism f , we write $\text{Sec}(f)$ and $\text{Ret}(f)$ for the types of sections and retractions of f respectively. We say f is a \mathbb{C} -*isomorphism* if we have an element of the type $\text{is-iso}_{\mathbb{C}}(f) := \text{Sec}(f) \times \text{Ret}(f)$ and write $x \cong_{\mathbb{C}} y$ for the type of isomorphisms between two objects $x, y: \mathbb{C}$.

⁵ There is no issue extending our results to multiple universes, but we only need one.

⁶ Capriotti and Kraus call this a *wild precategory*.

Remark 2.4 The isomorphisms in the wild category \mathcal{U} are exactly the categorical equivalences introduced in Section 1. To avoid confusion, we refer to a pair of functions $s: A \rightarrow B$ and $r: B \rightarrow A$ between types such that $rs \sim \text{id}_A$ as a *homotopy section* and *homotopy retraction* respectively. The term *isomorphism* is sometimes used in the literature to refer to what the HoTT Book [44] calls *quasi-inverses*, that is, maps $f: A \rightarrow B$ and $g: B \rightarrow A$ with homotopies $gf \sim \text{id}_A$ and $fg \sim \text{id}_B$. We never use the term in this way.

The following holds by a Yoneda style argument.

Lemma 2.5 *For a morphism $f: x \rightarrow y$ in a wild category \mathbb{C} , the following are logically equivalent:*

- (i) f is an isomorphism,
- (ii) $f^*: \mathbb{C}(x, z) \rightarrow \mathbb{C}(y, z)$ is an equivalence for all $z: \mathbb{C}$,
- (iii) $f_*: \mathbb{C}(z, y) \rightarrow \mathbb{C}(z, x)$ is an equivalence for all $z: \mathbb{C}$.

Lemma 2.6 *Given an isomorphism $f: x \rightarrow y$ in a wild category, the type $\text{Sec}(f)$ is contractible.*

Proof. Denote by f^{-1} the retraction of f . By Lemma 2.5 and [36, Exercise 9.1] we have the equivalences $(\sum_{g: y \rightarrow x} fg = \text{id}_y) \simeq (\sum_{g: y \rightarrow x} f^{-1}(fg) = f^{-1}\text{id}) \simeq (\sum_{g: y \rightarrow x} g = f^{-1})$. In the last step we use that composition with a path is an equivalence. The last type is contractible. \square

Corollary 2.7 *For a morphism f in a wild category, $\text{is-iso}(f)$ is a proposition.*

Proof. We show that $\text{is-iso}(f)$ is contractible, assuming that f is an isomorphism [36, Proposition 12.1.3]. By Lemma 2.6 and its dual, both $\text{Sec}(f)$ and $\text{Ret}(f)$ are contractible. \square

Lemma 2.8 *Isomorphisms in a wild category satisfy 2-out-of-3.*

Proof. By associativity we have $(fg)^* \sim f^*g^*$. The structure of an equivalence transfers across homotopies. Hence, the claim follows from 2-out-of-3 for equivalences [36, Exercise 9.4]. \square

For every object $x: \mathbb{C}$ in a wild category, the identity $\text{id}_x: x \rightarrow x$ is an isomorphism. By path induction, we may generalize to a map $\text{id-to-iso}: x =_{\mathbb{C}} y \rightarrow x \cong_{\mathbb{C}} y$ for $x, y: \mathbb{C}$. We define [11, Definition 4.16]:

Definition 2.9 A wild category \mathbb{C} is *univalent* if $\text{id-to-iso}: x =_{\mathbb{C}} y \rightarrow x \cong_{\mathbb{C}} y$ is an equivalence for $x, y: \mathbb{C}$.

Lemma 2.10 *A wild category \mathbb{C} is univalent exactly if $\sum_{y: \mathbb{C}} x \cong_{\mathbb{C}} y$ is contractible for all x .*

Proof. By the fundamental theorem of identity types [36, Theorem 11.2.2]. \square

Univalence of a universe \mathcal{U} ($\text{UA}_{\mathcal{U}}$, Definition 1.2) cannot be formulated on the level of an arbitrary wild category, as it refers to homotopy of functions. As $\text{UA}_{\mathcal{U}}$ implies $\text{FE}_{\mathcal{U}}$, however, it also implies that \mathcal{U} is a univalent wild category. Absent $\text{FE}_{\mathcal{U}}$, the converse may fail: as we will see, \mathcal{U} can be a univalent wild category without id-to-eq being an equivalence. We can consider ordinary univalence as the conjunction of two equivalences: one between $A =_{\mathcal{U}} B$ and $A \cong_{\mathcal{U}} B$ and one between $A \cong_{\mathcal{U}} B$ and $A \simeq B$.

2.2 Categorical equivalences and function extensionality

In comparing $A \cong_{\mathcal{U}} B$ and $A \simeq B$, it is natural to forget about universes entirely. Recall from Section 1 that a function $f: A \rightarrow B$ between (possibly large) types is a *categorical equivalence* if it admits a section and retraction, that is, $s, r: B \rightarrow A$ with $fs = \text{id}$ and $rf = \text{id}$. We write $\text{is-ceq}(f)$ for the type of witnesses that f is a categorical equivalence, and $A \cong B$ for the type of categorical equivalences from A to B .

In ITT, the only closed categorical equivalences are the strict isomorphisms, such as $A \times B \cong B \times A$. With $\text{CUA}_{\mathcal{U}}$ (Definition 1.3), there are more; for example, any $e: A \cong B$ in \mathcal{U} yields $(a =_A a') \cong (ea =_B ea')$ for $a, a': A$. The map $\text{ceq-to-eq}: (A \cong B) \rightarrow (A \simeq B)$, which converts equalities in function types to homotopies, becomes an equivalence under FE. In fact, it is an equivalence *only* if FE holds.

Definition 2.11 *Equivalence improvement (EI)* is the principle that for all types A, B , the map $\text{ceq-to-eq}: (A \cong B) \rightarrow (A \simeq B)$ is an equivalence.

We recall a lemma familiar from proofs that $\text{UA}_{\mathcal{U}}$ implies $\text{FE}_{\mathcal{U}}$ [44, Theorem 4.9.4] [36, Theorem 17.3.2]:

Lemma 2.12 *For any family of types $a: A \vdash B(a)$, the type $\prod_{a:A} B(a)$ is equivalent to the fiber of $\pi_*: (\sum_{a:A} B(a))^A \rightarrow A^A$ at $\text{id}_A: A^A$, where π_* is post-composition with the first projection. Equivalently, the following strictly commutative square is a homotopy pullback:*

$$\begin{array}{ccc} \prod_{a:A} B(a) & \xrightarrow{\langle \text{id}_A, - \rangle} & \left(\sum_{a:A} B(a) \right)^A \\ \downarrow & \lrcorner & \downarrow \pi_* \\ 1 & \xrightarrow{\text{id}_A} & A^A. \end{array}$$

Proof. We prove that the fiber of π_* at an arbitrary $t: A \rightarrow A$ is equivalent to $\prod_{a:A} B(ta)$. Define

$$\begin{aligned} s: (\prod_{a:A} B(ta)) &\rightarrow \text{fib}_{\pi_*}(t) & r: \text{fib}_{\pi_*}(t) &\rightarrow \prod_{a:A} B(ta) \\ f &\mapsto \langle \lambda a. \langle ta, fa \rangle, \text{refl} \rangle & \langle g, \text{refl} \rangle &\mapsto \pi_1 \circ g \end{aligned}$$

where π_1 is the second projection from $\sum_{a:A} B(a)$. We have $rs \doteq \text{id}$ and a homotopy $sr \sim \text{id}$ by path induction. \square

Theorem 2.13 *In ITT, the following are logically equivalent:*

- (i) El: for all types A, B , $\text{ceq-to-eq}: (A \cong B) \rightarrow (A \simeq B)$ is an equivalence,
- (ii) for all types A, B , $\text{ceq-to-eq}: (A \cong B) \rightarrow (A \simeq B)$ admits a homotopy section,
- (iii) for all types A, B and every $f: A \rightarrow B$, the type $\text{is-equiv}(f)$ is a proposition,
- (iv) for every type A and $f: A \rightarrow A$, if $f \sim \text{id}_A$ then $f = \text{id}_A$,
- (v) for all types A, B and every $f: A \rightarrow B$, we have $\text{is-equiv}(f) \rightarrow \text{is-ceq}(f)$,
- (vi) for all types A, B and every $f: A \rightarrow B$, we have $\text{is-equiv}(f) \rightarrow \text{is-equiv}(f_*)$,
- (vii) Weak FE: for every family of contractible types $a: A \vdash P(a)$, the type $\prod_{a:A} P(a)$ is contractible,
- (viii) FE: for every $a: A \vdash B(a)$ and $f, g: \prod_{a:A} B(a)$, the map $(f = g) \rightarrow (f \sim g)$ is an equivalence.

Proof. That (i) \implies (ii) is immediate. For (ii) \implies (iii), note that any homotopy section of ceq-to-eq exhibits $\text{is-equiv}(f)$ as a homotopy retract of the proposition $\text{is-ceq}(f)$ (cf. Corollary 2.7). For (iii) \implies (iv), observe that a homotopy $f \sim \text{id}_A$ implies that f is a homotopy inverse of id_A . As id_A is also its own homotopy inverse, (iii) implies that $f = \text{id}_A$. That (iv) \implies (v) is immediate, and (v) \implies (vi) follows from Lemma 2.5. The implication (vi) \implies (vii), which appears in standard proofs of $\text{FE}_{\mathcal{U}}$ from $\text{UA}_{\mathcal{U}}$ [44, Theorem 4.9.4] [36, Theorem 17.3.2], follows from Lemma 2.12 and the fact that the fibers of an equivalence are contractible [36, Theorem 10.4.6]. That (vii) \implies (viii) is due to Voevodsky; see for example [44, Theorem 4.9.5] or [36, Theorem 13.1.2]. That (viii) \implies (i) is by definition of ceq-to-eq . \square

Theorem 2.13 relativizes to \mathcal{U} . From this we recover that $\text{UA}_{\mathcal{U}}$ implies $\text{FE}_{\mathcal{U}}$ (and thus $\text{CUA}_{\mathcal{U}}$).

Corollary 2.14 $\text{ITT} \vdash \text{UA}_{\mathcal{U}} \leftrightarrow (\text{CUA}_{\mathcal{U}} \wedge \text{FE}_{\mathcal{U}})$.

Proof. For $A, B: \mathcal{U}$, consider the following homotopy commutative triangle.

$$\begin{array}{ccc} & (A \cong B) & \\ \text{id-to-ceq} \nearrow & & \text{ceq-to-eq} \searrow \\ (A =_{\mathcal{U}} B) & \xrightarrow{\text{id-to-eq}} & (A \simeq B) \end{array}$$

The map ceq-to-eq is an equivalence for all $A, B: \mathcal{U}$ if and only if $\text{FE}_{\mathcal{U}}$ holds, by (i) \iff (viii) of Theorem 2.13 in \mathcal{U} . Thus, if both $\text{CUA}_{\mathcal{U}}$ and $\text{FE}_{\mathcal{U}}$ hold, then $\text{UA}_{\mathcal{U}}$ holds by 2-out-of-3 for equivalences.

Conversely, if $\text{UA}_{\mathcal{U}}$ holds, then id-to-eq has in particular a homotopy section for all $A, B: \mathcal{U}$. Post-composing these homotopy sections with id-to-ceq yields homotopy sections of ceq-to-eq for all $A, B: \mathcal{U}$.

By (ii) \implies (viii), (i) of Theorem 2.13 relativized to \mathcal{U} , this implies that $\text{FE}_{\mathcal{U}}$ holds and that ceq-to-eq is an equivalence for all $A, B: \mathcal{U}$. From the latter, $\text{CUA}_{\mathcal{U}}$ follows by 2-out-of-3. \square

The usual proof of $\text{ITT} + \text{UA}_{\mathcal{U}} \vdash \text{FE}_{\mathcal{U}}$ thus factors through a universe-independent proof of $\text{ITT} + \text{EI} \vdash \text{FE}$. We now show that this decomposition of $\text{UA}_{\mathcal{U}}$ is proper: neither $\text{CUA}_{\mathcal{U}}$ nor $\text{FE}_{\mathcal{U}}$ alone implies $\text{UA}_{\mathcal{U}}$. That $\text{ITT} + \text{FE}_{\mathcal{U}} \not\vdash \text{UA}_{\mathcal{U}}$ follows from the standard model of type theory in \mathbf{Set} , so it is $\text{ITT} + \text{CUA}_{\mathcal{U}} \not\vdash \text{UA}_{\mathcal{U}}$ that we need to establish. Equivalently, we want to show that $\text{ITT} + \text{CUA}_{\mathcal{U}} \not\vdash \text{FE}_{\mathcal{U}}$.

3 Von Glehn’s polynomial model construction

We use Von Glehn’s *polynomial model* construction [50,33] to separate $\text{UA}_{\mathcal{U}}$ from $\text{CUA}_{\mathcal{U}}$. This model construction is related [50, §5.2] [32, §7.1] to Gödel’s Dialectica interpretation [23], versions of which have been used to interpret various logics; see for example de Paiva’s categorical models of linear logics [17]. For our purposes, the relevant feature of the polynomial models is that they always refute FE .

By a model of type theory, we mean a category with attributes [13], category with families [19], or natural model [6]: a presheaf of types $\text{Ty}_{\mathbb{C}}: \mathbb{C}^{\text{op}} \rightarrow \mathbf{Set}$ and presheaf of terms $\text{Tm}_{\mathbb{C}}: (\int_{\mathbb{C}} \text{Ty}_{\mathbb{C}})^{\text{op}} \rightarrow \mathbf{Set}$. Here we depart from Von Glehn, who works with non-split models (categories with display maps), instead following Kovács’ Agda formalization of a version of the construction [30].⁷ We therefore describe the model in detail below, referencing Von Glehn’s analogous construction for each component. We leave it to the reader to translate some verifications to the split case.

We fix a base model \mathbb{C} . For $\sigma: \Delta \rightarrow \Gamma$, we write the action of σ on $A \in \text{Ty}_{\mathbb{C}}(\Gamma)$ and $a \in \text{Tm}_{\mathbb{C}}(\Gamma, A)$ as $A\sigma$ and $a\sigma$ respectively. For $a \in \text{Tm}_{\mathbb{C}}(\Gamma, A)$, we write the induced substitution as $[a]: \Gamma \rightarrow \Gamma.A$. For $\Gamma \in \mathbb{C}$, the category $\mathbf{Ty}_{\mathbb{C}}(\Gamma)$ has objects $\text{Ty}_{\mathbb{C}}(\Gamma)$ and morphisms $A \rightarrow B$ given by $(\mathbb{C}/\Gamma)(\mathfrak{p}_A, \mathfrak{p}_B)$, i.e., morphisms $\Gamma.A \rightarrow \Gamma.B$ over Γ , i.e, elements of $\text{Tm}(\Gamma.A, B\mathfrak{p})$. This extends to a functor $\mathbf{Ty}_{\mathbb{C}}: \mathbb{C}^{\text{op}} \rightarrow \mathbf{Cat}$. For ease of readability, we sometimes give variable names to context extensions, in which case we write $\Gamma, a: A$ instead of $\Gamma.A$.

Definition 3.1 The category $\mathbf{Poly}(\mathbb{C})$ is given by $\int_{\Gamma \in \mathbb{C}} \mathbf{Ty}_{\mathbb{C}}(\Gamma)^{\text{op}}$, the Grothendieck construction of the fiberwise opposite of $\mathbf{Ty}_{\mathbb{C}}$. For $\Gamma \in \mathbf{Poly}(\mathbb{C})$, we set $(\Gamma_S, \Gamma_P) := \Gamma$ and refer to the two components as *shapes* and *positions* respectively. A morphism $\sigma: \Delta \rightarrow \Gamma$ is given by some $\sigma_S: \Delta_S \rightarrow \Gamma_S$ in \mathbb{C} and a morphism $\sigma_P: \Gamma_P \sigma_S \rightarrow \Delta_P$ in $\mathbf{Ty}_{\mathbb{C}}(\Delta_S)$ [21, (14)]. Composition of $\sigma: \Theta \rightarrow \Delta$ and $\tau: \Delta \rightarrow \Gamma$ is given by $(\sigma \circ \tau)_S = \sigma_S \circ \tau_S$ and $(\sigma \circ \tau)_P = \tau_P \circ \sigma_P \tau_S$, where $\sigma_P \tau_S: \Theta_P \sigma_S \tau_S \rightarrow \Delta_P \tau_S$.

Remark 3.2 Suppose \mathbb{C} is a democratic model of extensional type theory [14, Definition 2.6], so $\mathbf{Ty}_{\mathbb{C}}(\Gamma)$ is naturally equivalent to \mathbb{C}/Γ . Then $\mathbf{Poly}(\mathbb{C})$ is the *category of (single-variable) polynomials in \mathbb{C}* , denoted $\text{Poly}_{\mathbb{C}}(1, 1)$ by Gambino and Kock [21, §2-14]. $\mathbf{Poly}(\mathbb{C})$ is called the *category of containers* in a line of work starting with Abbott, Altenkirch, and Ghani [2,1]. However, the *container model* sketched by Altenkirch and Kaposi [3] is distinct from the polynomial model; it has the same category of contexts but more types.

Definition 3.3 The presheaf of types $\text{Ty}_{\mathbf{Poly}(\mathbb{C})}$ is given by restricting the presheaf $\sum_{A: \text{Ty}_{\mathbb{C}}} \text{Ty}^{\text{Tm}(A)}$ on \mathbb{C} along the projection $\mathbf{Poly}(\mathbb{C}) \rightarrow \mathbb{C}$. This means an element of $\text{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ is given by some $A_S \in \text{Ty}_{\mathbb{C}}(\Gamma_S)$ and $A_P \in \text{Ty}_{\mathbb{C}}(\Gamma_S.A_S)$, which we respectively refer to as the *shapes* and *positions* of A .

To make $\mathbf{Poly}(\mathbb{C})$ into a model, we need \mathbb{C} to have finite coproducts of types satisfying the strict η rule. This is used in the substitution calculus (Proposition 3.6). We can break these up into binary ($A_0 + A_1$) and nullary (0) coproducts. We write the eliminator for binary coproducts as follows.

$$\frac{\Gamma, v: A_0 + A_1, \Delta(v) \vdash P(v) \quad \text{for } i \in \{0, 1\}: \Gamma, a_i: A_i, \Delta(\text{in}_i(a_i)) \vdash u_i: P(\text{in}_i(a_i))}{\Gamma, v: A_0 + A_1, \Delta(v) \vdash \text{elim}_+^P(a_0.u_0, a_1.u_1, v): P(v)}$$

⁷ Kovács constructs $\mathbf{Poly}(\mathcal{U})$ where \mathcal{U} is the model of type theory internal to Agda associated to some universe. Given a model \mathbb{A} of Agda’s type theory, interpreting the formalization in \mathbb{A} yields the polynomial model whose base model is the interpretation of \mathcal{U} in \mathbb{A} . Kovács postulates uniqueness of identity proofs and FE , but essentially only to deal with the mismatch between Agda’s coproducts and the strict coproducts required for Von Glehn’s construction. Unfortunately, uniqueness of identity proofs is inconsistent with $\text{CUA}_{\mathcal{U}}$, so we cannot build directly on this formalization.

The β rules state that $\Gamma, a_i : A_i, \Delta(\text{in}_i(a_i)) \vdash \text{elim}_+^P(a_0.u_0, a_1.u_1, \text{in}_i(a_i)) \doteq a_i$. The η rule states, as shown below, that we can test strict equality of terms depending on $A_0 + A_1$ by checking equality on constructors.

$$\frac{\Gamma, u : A_0 + A_1, \Delta \vdash t(u), t'(u) : P(u) \quad \text{for } i \in \{0, 1\}: \Gamma, a : A_i, \Delta(\text{in}_i(a)) \vdash t(\text{in}_i(a)) \doteq t'(\text{in}_i(a)) : P(\text{in}_i(a))}{\Gamma, u : A_0 + A_1, \Delta \vdash t(u) \doteq t'(u) : P(u)}$$

The elimination rule and η law for 0 are similar.

$$\frac{\Gamma, v : 0, \Delta(v) \vdash P(v)}{\Gamma, v : 0, \Delta(v) \vdash \text{elim}_0^P(v) : P(v)} \quad \frac{\Gamma, v : 0, \Delta(v) \vdash t(v), t'(v) : P(v)}{\Gamma, v : 0, \Delta(v) \vdash t(v) \doteq t'(v) : P(v)}$$

Semantically, the above rules mean that the split fibration $\mathbf{Ty}_{\mathbb{C}}$ has split fibred coproducts [27, Definition 1.8.1]: each $\mathbf{Ty}_{\mathbb{C}}(\Gamma)$ has chosen finite coproducts and the substitution functors $\mathbf{Ty}(\Gamma) \rightarrow \mathbf{Ty}(\Delta)$ preserve them strictly. For inference rules, see for example Von Glehn [50, §2.3.5] or Angiuli and Gratzer [5, §2.5.1 and §2.5.3].

Remark 3.4 Strict η laws for coproducts are often omitted from the syntax of type theory due to issues with strict equality checking. In the simply-typed λ -calculus, strict equality checking for coproducts with the η law is decidable but difficult. Ghani [22] addresses the case of binary coproducts; Scherer [37] handles the empty type. In ITT , the η law for the empty type makes strict equality undecidable: with this law, deciding whether $a : A \vdash \text{in}_0(\star) \doteq \text{in}_1(\star) : 1 + 1$ requires deciding whether A implies 0. To our knowledge, it is an open problem whether strict equality is decidable for ITT with binary coproducts and their η law; see, for example, discussion between Shulman, Kovács, and others on Proof Assistants StackExchange [40].

We further require that our coproducts are *extensive*. The syntactic counterpart is often called *large elimination*. This means that given types $\Gamma, a_i : A_i, \Delta(\text{in}_i(a_i)) \vdash P_i(a_i)$ for $i \in \{0, 1\}$, there is a type $\Gamma, u : A_0 + A_1, \Delta(u) \vdash [P_0, P_1](u)$ satisfying $\Gamma, a_i : A_i, \Delta(\text{in}_i(a_i)) \vdash [P_0, P_1](\text{in}_i(a_i)) \doteq P_i(a_i)$. Using the strict η rule, for every family $\Gamma.A_0 + A_1 \vdash P$ there is a canonical strict isomorphism $P \cong [P\text{in}_0, P\text{in}_1]$. Furthermore, for all such P there is a strict isomorphism $\sum_{u:A+B} P(u) \cong \sum_{a:A} P(\text{in}_0(a)) + \sum_{b:B} P(\text{in}_1(b))$. This is used in particular in the construction of dependent product types (Proposition 3.12).

For the rest of this section, we assume \mathbb{C} has extensive finite coproducts of types with the strict η rule.

Definition 3.5 The presheaf of terms $\text{Tm}_{\mathbf{Poly}(\mathbb{C})} : (f \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})})^{\text{op}} \rightarrow \mathbf{Set}$ is given by $\text{Tm}_{\mathbf{Poly}(\mathbb{C})}(\Gamma, A) := \sum_{a_S : \text{Tm}(\Gamma_S, A_S)} \mathbf{Ty}(\Gamma_S)(A_P[a_S], \Gamma_P)$. We refer to the components again as *shapes* and *positions*.

Proposition 3.6 ($\mathbf{Poly}(\mathbb{C}), \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})}, \text{Tm}_{\mathbf{Poly}(\mathbb{C})}$) *extends to a category with families by setting*

$$\begin{aligned} (\Gamma.A)_S &:= \Gamma_S.A_S, & (\Gamma.A)_P &:= \Gamma_P \mathbf{p}_{A_S} + A_P, & (\mathbf{p}A)_S &:= \mathbf{p}_{A_S}, & (\mathbf{p}A)_P &:= \text{in}_0 : \Gamma_P \rightarrow \Gamma_P \mathbf{p} + A_P, \\ (\mathbf{q}A)_S &:= \mathbf{q}_{A_S}, & (\mathbf{q}A)_P &:= \text{in}_1 : A_P \rightarrow \Gamma_P \mathbf{p} + A_P, & \langle \sigma, a \rangle_S &:= \langle \sigma_S, a_S \rangle, & \langle \sigma, a \rangle_P &:= [\sigma_P, a_P]. \end{aligned}$$

Proof. Given $\sigma : \Delta \rightarrow \Gamma$ and $a \in \text{Tm}(\Delta, A\sigma)$ we have $\langle \sigma, a \rangle_S : \Delta_S \rightarrow \Gamma_S.A_S$ and $\langle \sigma, a \rangle_P : \Gamma_P \sigma_S + A_P \langle \sigma_S, a_S \rangle \rightarrow \Delta_P$ in $\mathbf{Ty}(\Delta)$. Clearly, all desired equations hold on shapes since they hold in \mathbb{C} . We have $(\mathbf{p}\langle \sigma, a \rangle)_P = [\sigma_P, a_P] \circ \text{in}_0 \mathbf{p} = \sigma_P$ and $(\mathbf{q}\langle \sigma, a \rangle)_P = [\sigma_P, a_P] \circ \text{in}_1 \mathbf{p} = a_P$. This shows $\mathbf{p}\langle \sigma, a \rangle = \sigma$ and $\mathbf{q}\langle \sigma, a \rangle = a$. Furthermore, $\langle \mathbf{p}, \mathbf{q} \rangle_P = [\mathbf{p}_P, \mathbf{q}_P] = [\text{in}_0, \text{in}_1] = \text{id}$ and $(\langle \sigma, a \rangle \tau)_P = \tau_P \circ [\sigma_P, a_P]_{\tau_S} = [\tau_P \circ \sigma_P \tau_S, \tau_P \circ a_P \tau_S] = \langle \sigma \tau, a \tau \rangle_P$. This shows $\langle \mathbf{p}, \mathbf{q} \rangle = \text{id}$ and $\langle \sigma, a \rangle \tau = \langle \sigma \tau, a \tau \rangle$. \square

3.1 Type formers

We give the interpretations in $\mathbf{Poly}(\mathbb{C})$ of Σ , identity, Π , binary and nullary coproduct, and universe types.

Proposition 3.7 (cf. [50, §4.2]) *Given $A \in \text{Ty}_{\text{Poly}(\mathbb{C})}(\Gamma)$, $B \in \text{Ty}_{\text{Poly}(\mathbb{C})}(\Gamma.A)$, we have a dependent sum $\Sigma_A B \in \text{Ty}(\Gamma)$ given by*

$$\Gamma_S \vdash (\Sigma_A B)_S := \sum_{a:A_S} B_S(a), \quad \Gamma_S, \langle a_S, b_S \rangle : (\Sigma_A B)_S \vdash (\Sigma_A B)_P := A_P(a_S) + B_P(a_S, b_S).$$

Proposition 3.8 (cf. [50, §4.4]) *Given $A \in \text{Ty}_{\text{Poly}(\mathbb{C})}(\Gamma)$ and $u, v \in \text{Tm}(\Gamma, A)$, we have an identity type $u =_A v \in \text{Ty}(\Gamma)$ given by*

$$\Gamma_S \vdash (u =_A v)_S := (u_S =_{A_S} v_S), \quad \Gamma_S, p : (u =_A v)_S \vdash (u =_A v)_P := 0,$$

with, for $u \in \text{Tm}(\Gamma, A)$, the reflexive path $\text{refl}_u \in \text{Tm}(\Gamma, u =_A u)$ given by

$$\Gamma_S \vdash (\text{refl}_u)_S := \text{refl}_{u_S} : (u_S =_{A_S} u_S), \quad \Gamma_S.0 \vdash (\text{refl}_u)_P := \text{elim}_0 : \Gamma_P.$$

Proof. For every $u \in \text{Tm}(\Gamma, A)$, $B \in \text{Ty}(\Gamma.A.u = \mathbf{q})$, $w \in \text{Tm}(\Gamma, B[u, \text{refl}_u])$, $v \in \text{Tm}(\Gamma, A)$, and $p \in \text{Tm}(\Gamma, u =_A v)$, the eliminator $\text{elim}_{=}^{B,u}(w, v, p) \in \text{Tm}(\Gamma, B[v, p])$ is given by

$$\begin{aligned} \Gamma_S & \vdash \text{elim}_{=}^{B,u}(w, v, p)_S := \text{elim}_{=}^{B_S, u_S}(w_S, v_S, p_S) : B_S[v_S, p_S], \\ \Gamma_S, b : B_P[v_S, p_S, \text{elim}_{=}^{B,u}(w, v, p)_S] & \vdash \text{elim}_{=}^{B,u}(w, v, p)_P := w_P(p_* b) : \Gamma_P, \end{aligned}$$

where $\Gamma_S.B_P[u_S, \text{refl}_{u_S}, w_S] \vdash w_P : \Gamma_P$ and $p_* : B_P[v_S, p_S, \text{elim}_{=}^{B,u}(w, v, p)_S] \rightarrow B_P[u_S, \text{refl}_{u_S}, w_S]$ is defined by path induction on p_S . The β rule for $\text{elim}_{=}$ follows from the same β rule in the base model. \square

Remark 3.9 Von Glehn takes $(u =_A v)_P$ to be the constant family $A_P(u_S) + A_P(v_S)$. We follow Kovács [30] by instead taking the constant family 0, which simplifies our arguments. Since both definitions satisfy the rules of the identity type, they are equivalent—though not categorically equivalent. The equivalence suffices to imply that our main result Theorem 4.17 transfers to Von Glehn’s identity types. This kind of flexibility is common to type formers without strict η laws in the polynomial model.

We treat the construction of Π types in more detail. Here, we depend crucially on extensivity of coproducts. First, we define families over $A_0 + A_1$ that are inhabited over exactly one of the inclusions.

Definition 3.10 For types A_0, A_1 , define the families $A_0 + A_1 \vdash \text{is}_0 := [1, 0]$ and $A_0 + A_1 \vdash \text{is}_1 := [0, 1]$.

Lemma 3.11 For types A_0, A_1 , the families $A_0 + A_1 \vdash \text{is}_0, \text{is}_1$ are strict propositions.

Proof. More generally, if $A_i \vdash P_i$ are strict propositions for $i \in \{0, 1\}$ then so is $[P_0, P_1]$: we have $u : A_0 + A_1, v_0, v_1 : [P_0, P_1] \vdash v_0 \overset{\circ}{=} v_1$ directly from the strict η rule since $[P_0, P_1](\text{in}_i(u_i)) \overset{\circ}{=} P_i(u_i)$. \square

Proposition 3.12 (cf. [50, §4.3]) *Given $A \in \text{Ty}_{\text{Poly}(\mathbb{C})}(\Gamma)$, $B \in \text{Ty}_{\text{Poly}(\mathbb{C})}(\Gamma.A)$, we have a dependent product $\Pi_A B \in \text{Ty}(\Gamma)$ given by*

$$\begin{aligned} \Gamma_S & \vdash (\Pi_A B)_S := \sum_{f_S : \prod_{a:A_S} B_S(a)} \prod_{a:A_S} B_P(a, f_S(a)) \rightarrow 1 + A_P(a), \\ \Gamma_S, \langle f_{SS}, f_{SP} \rangle : (\Pi_A B)_S & \vdash (\Pi_A B)_P := \sum_{\substack{a:A_S \\ b:B_P(a, f_{SS}(a))}} \text{is}_0(f_{SP}(a, b)). \end{aligned}$$

For $f \in \text{Tm}(\Gamma, \Pi_A B)$ we write f_{SS} and f_{SP} for the first and second component of f_S respectively.

Proof. It suffices to define a natural isomorphism $\lambda : \text{Tm}(\Gamma.A, B) \cong \text{Tm}(\Gamma, \Pi_A B) : \text{app}$. By Definition 3.5, an element of $\text{Tm}(\Gamma.A, B)$ corresponds to a pair

$$\Gamma_S \vdash b_S : \prod_{a:A_S} B_S(a), \quad \Gamma_S \vdash b_P : \prod_{a:A_S} B_P(a, b_S(a)) \longrightarrow \Gamma_P + A_P(a).$$

By Definition 3.5 and the curry-uncurry isomorphism, an element of $\text{Tm}(\Gamma, \Pi_A B)$ corresponds to a triple

$$\begin{aligned} \Gamma_S \vdash f_{SS} &: \prod_{a:A_S} B_S(a), & \Gamma_S \vdash f_{SP} &: \prod_{a:A_S} B_P(a, f_{SS}(a)) \longrightarrow 1 + A_P(a), \\ \Gamma_S \vdash f_P &: \prod_{a:A_S} \left(\sum_{b:B_P(a, f_{SS}(a))} \text{is}_0(f_{SP}(a, b)) \right) \longrightarrow \Gamma_P. \end{aligned}$$

Now, note that we have for all types X, Y, Z that

$$\sum_{f:X \rightarrow 1+Y} \prod_{x:X} Z^{\text{is}_0(fx)} \cong \prod_{x:X} \sum_{u:1+Y} Z^{\text{is}_0(u)} \cong \prod_{x:X} \left(\sum_{\star:1} Z^{\text{is}_0(\text{in}_0(\star))} + \sum_{y:Y} Z^{\text{is}_0(\text{in}_1(y))} \right) \cong (Z + Y)^X.$$

Applying this strict isomorphism to the above yields the desired bijection. \square

Proposition 3.13 (cf. [50, §4.2]) *We have a nullary coproduct $0 \in \text{Ty}_{\mathbf{Poly}(\mathbb{C})}$ and binary coproduct $A + B \in \text{Ty}(\Gamma)$ for $A, B \in \text{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ given by*

$$\begin{aligned} \Gamma_S &\vdash (A + B)_S := A_S + B_S, & \Gamma_S &\vdash 0_S := 0, \\ \Gamma_S.(A_S + B_S) &\vdash (A + B)_P := [A_P, B_P], & \Gamma_S.0 &\vdash 0_P := \text{elim}_0. \end{aligned}$$

These satisfy the strict η rule and are extensive.

Proposition 3.14 (cf. [32, Proposition 7.1.6]) *We have a universe $\mathcal{U} \in \text{Ty}_{\mathbf{Poly}(\mathbb{C})}(1)$ with decoding function $\text{El} \in \text{Ty}_{\mathbf{Poly}(\mathbb{C})}(1, \mathcal{U})$ (which we usually leave implicit) given by*

$$\begin{aligned} 1 &\vdash \mathcal{U}_S := \sum_{A_S:\mathcal{U}} \mathcal{U}^{A_S}, & 1, \langle A_S, A_P \rangle : \mathcal{U}_S &\vdash \text{El}_S := A_S, \\ 1, \langle A_S, A_P \rangle : \mathcal{U}_S &\vdash \mathcal{U}_P := 0, & 1, \langle A_S, A_P \rangle : \mathcal{U}_S, a_S : \text{El}_S \langle A_S, A_P \rangle &\vdash \text{El}_P := A_P(a_S). \end{aligned}$$

3.2 A dependent right adjoint

The category \mathbb{C} is a reflective subcategory of $\mathbf{Poly}(\mathbb{C})$: the functor $-_S: \mathbf{Poly}(\mathbb{C}) \rightarrow \mathbb{C}$ has a fully faithful right adjoint $\circ_S: \mathbb{C} \rightarrow \mathbf{Poly}(\mathbb{C})$ given on objects by $\circ_S \Gamma := (\Gamma, 0)$. Von Glehn [50] uses this adjunction to define the model that we defined manually above, transferring it from \mathbb{C} . Both functors extend to pseudomorphisms of models in the sense of Kaposi, Huber, and Sattler [28, §4].

Lemma 3.15 *The adjunction $\circ_S: \mathbb{C} \rightleftarrows \mathbf{Poly}(\mathbb{C}) : -_S$ lifts to an adjunction of pseudomorphisms of models, with left adjoint projecting to shapes of types and terms, and the right adjoint given on types and terms by $\circ_S A := (A, 0)$ and $\circ_S a := (a, \text{elim}_0)$.*

Proof. Immediate from the definition of context extension and that $0 + 0 \cong 0$. \square

The right adjoint morphism induces a dependent right adjoint (cf. [25, §7]).

Corollary 3.16 *The operation $\text{Ty}(\Gamma_S) \rightarrow \text{Ty}(\Gamma), A \mapsto (\circ_S A)\eta_\Gamma$ defines a dependent right adjoint.*

Proof. $\text{Tm}(\Gamma, (\circ_S A)\eta_\Gamma) \cong (\mathbf{Poly}(\mathbb{C})/\circ_S \Gamma)(\Gamma, \circ_S \Gamma_S. \circ_S A) \cong (\mathbb{C}/\Gamma_S)(\Gamma_S, \Gamma_S.A) \cong \text{Tm}(\Gamma_S, A)$. \square

Henceforth, we denote by \circ_S the dependent right adjoint, not the morphism. The composite mapping $A \mapsto \circ_S(A_S)$ defines a pointed endofunctor on $\text{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ (and a lex operation in the sense of [16, Remark 5]). If clear from context, we write just $\circ_S A$ for this composite.

4 Familial categorical univalence in the polynomial model

Fix an input model \mathbb{C} as in Section 3. To study $\mathbf{CUA}_{\mathcal{U}}^{\bullet}$ in $\mathbf{Poly}(\mathbb{C})$, we analyze the wild category \mathcal{U}^I and its isomorphisms. To simplify calculations, we redefine here $\mathcal{U}^I(A, B) := \prod_{u: \sum_I A} B(\pi_0 u)$. This is strictly isomorphic to the type $\prod_{i:I} A(i) \rightarrow B(i)$ in Definition 1.4, so the two versions of $A \cong_{\mathcal{U}^I} B$ are related by an equivalence preserving the identity isomorphism up to path. Hence, $\mathbf{CUA}_{\mathcal{U}}^{\bullet}$ is invariant under this change.

We start by unfolding the type $\mathcal{U}^I(A, B)$ and composition in \mathcal{U}^I . A key observation is that the shape part of $f \in \mathbf{Tm}(\Gamma, A \rightarrow B)$ consists of a function between shapes and a *partial* function between positions.

Lemma 4.1 *For $\Gamma \in \mathbf{Poly}(\mathbb{C})$, $I \in \mathbf{Ty}(\Gamma)$, and $A, B \in \mathbf{Ty}(\Gamma.I)$, the type $\mathcal{U}^I(A, B) \in \mathbf{Ty}(\Gamma)$ is given by*

$$\begin{aligned} \Gamma_S \quad \vdash \mathcal{U}^I(A, B)_S &\doteq \sum_{f_S: \mathcal{U}^{I_S}(A_S, B_S)} \prod_{\substack{i: I_S \\ a: A_S(i)}} B_P(i, f_S(i, a)) \rightarrow 1 + (I_P(i) + A_P(i, a)), \\ \Gamma_S, \langle f_S, f_P \rangle: \mathcal{U}^I(A, B)_S \vdash \mathcal{U}^I(A, B)_P &\doteq \sum_{\substack{i: I_S, a: A_S(i) \\ b: B_P(i, a, f_S(a))}} \text{is}_0(f_P(i, a, b)) \end{aligned}$$

Proof. Direct unfolding using Propositions 3.7 and 3.12. \square

Lemma 4.2 *If $f \in \mathbf{Tm}(\Gamma, \mathcal{U}^I(B, C))$, $g \in \mathbf{Tm}(\Gamma, \mathcal{U}^I(A, B))$, then the composite $fg \in \mathbf{Tm}(\Gamma, \mathcal{U}^I(A, C))$ is given by*

$$\begin{aligned} \Gamma_S \vdash (fg)_{SS} &\doteq f_{SS} \circ g_{SS}: \mathcal{U}^{I_S}(A_S, C_S), \\ \Gamma_S \vdash (fg)_{SP} &\doteq \lambda i. \lambda a. [\text{in}_0, \text{in}_0, g_{SP}(i, a)] \circ f_{SP}(g_{SS}(i, a)): \prod_{\substack{i: I_S \\ a: A_S(i)}} C_P(i, (fg)_{SS}(a)) \rightarrow 1 + (I_P(i) + A_P(i, a)). \end{aligned}$$

Proof. We have that $\mathbf{Tm}(\Gamma, \mathcal{U}^I(A, B)) \cong \mathbf{Tm}(\Gamma.I.A, B_P)$. Let $u \in \mathbf{Tm}(\Gamma.I.B, C_P)$ and $v \in \mathbf{Tm}(\Gamma.I.A, B_P)$ given by $u_S \in \mathbf{Tm}(\Gamma_S.I_S.B_S, C_S P)$, $v_S \in \mathbf{Tm}(\Gamma_S.I_S.A_S, B_S P)$, $u_P: C_P(\mathbf{p}, u_S) \rightarrow (\Gamma_P + I_P)\mathbf{p} + B_P$ in $\mathbf{Ty}(\Gamma_S.I_S.B_S)$, and $v_P: B_P(\mathbf{p}, v_S) \rightarrow (\Gamma_P + I_P)\mathbf{p} + A_P$ in $\mathbf{Ty}(\Gamma_S.I_S.A_S)$. The composite of u and v is by definition $u\langle \mathbf{p}, v \rangle \in \mathbf{Tm}(\Gamma.I.A, C_P)$. Direct calculation using Definition 3.5 and Proposition 3.6 shows that $(u\langle \mathbf{p}, v \rangle)_S = u_S\langle \mathbf{p}, v_S \rangle$ and $(u\langle \mathbf{p}, v \rangle)_P = [\text{in}_0, v_P] \circ u_P\langle \mathbf{p}, v_S \rangle: C_P u_S\langle \mathbf{p}, v_S \rangle \rightarrow (\Gamma_P + I_P)\mathbf{p} + A_P$. Composing with the λ -app bijection from Proposition 3.12 yields the desired description. \square

4.1 Categories of partial functions

We now introduce an auxiliary wild category in \mathbb{C} . It can be viewed as the Kleisli category of the monad on \mathcal{U}^I given by coproduct with a fixed family $J: I \rightarrow \mathcal{U}$, though we will not explicitly develop this viewpoint. To see that this even is a wild category in our setting, we rely on the strict properties of coproducts.

Proposition 4.3 (In \mathbb{C}) *For every family $J: I \rightarrow \mathcal{U}$, the following defines a wild category \mathcal{U}_J^I :*

$$(\mathcal{U}_J^I)_0 := \mathcal{U}^I, \quad (\mathcal{U}_J^I)_1(A, B) := \prod_{i:I} A(i) \rightarrow J(i) + B(i), \quad (\text{id}_A)_i := \text{in}_1, \quad (f \circ g)_i := [\text{in}_0, f_i] \circ g_i,$$

with unitors and associators given by reflexivity.

Proof. Direct calculation using the η rules for Π and $+$. \square

Morphisms in \mathcal{U}_J^I can be thought of as families of partial functions, with J as a type of “errors”. We introduce a notion of *total* morphism in \mathcal{U}_J^I . By η for coproducts, total morphisms coincide with morphisms in \mathcal{U}^I up to equivalence. Crucially, all isomorphisms in \mathcal{U}_J^I will be total.

Definition 4.4 (In \mathbb{C}) A morphism $f: \mathcal{U}_J^I(A, B)$ is *total* if $\text{is-tot}(f) := \prod_{i:I, a:A(i)} \text{is}_1(f_i a)$ is inhabited. We define $\mathcal{U}_{J, \text{tot}}^I(A, B) := \sum_{f: \mathcal{U}_J^I(A, B)} \text{is-tot}(f)$.

Lemma 4.5 (In \mathbb{C}) For all $f: \mathcal{U}_J^I(A, B)$, the type $\text{is-tot}(f)$ is a homotopy proposition.

Proof. $\Pi_C P$ is a strict proposition if P is: for $p, q: \Pi_C P$ we have $p \doteq \lambda x.p(x) \doteq \lambda y.q(y) \doteq q$. Hence, it follows from Lemma 3.11 that $\text{is-tot}(f)$ is even a strict proposition. \square

Lemma 4.6 (In \mathbb{C}) For all types I and $I \vdash J, B$, the map $(\prod_{i:I} B(i)) \rightarrow \prod_{i:I} \sum_{u:J(i)+B(i)} \text{is}_1(u)$ given by $f \mapsto \lambda i.(\text{in}_1(f_i u), \star)$ is an equivalence.

Proof. Π type formation sends families of strict isomorphisms to strict isomorphisms. For $i: I$ we have

$$B(i) \cong \left(\sum_{j:J(i)} 0 \right) + \left(\sum_{b:B(i)} 1 \right) \cong \left(\sum_{j:J(i)} \text{is}_1(\text{in}_0(j)) \right) + \left(\sum_{b:B(i)} \text{is}_1(\text{in}_1(b)) \right) \cong \sum_{u:J(i)+B(i)} \text{is}_1(u).$$

In each step we use that to check the commutation out of a coproduct, it suffices to check after precomposing with both inclusions. \square

Corollary 4.7 (In \mathbb{C}) For all types I and $J, A, B: I \rightarrow \mathcal{U}$, the map $\mathcal{U}^I(A, B) \rightarrow \mathcal{U}_{J,\text{tot}}^I(A, B)$ given by $f \mapsto (\text{in}_1 \circ f, \lambda i.\lambda a.\star)$ is an equivalence.

Proof. Instantiate Lemma 4.6 with index type $\sum_{i:I} A$ and the families $(i, a): \sum_I A \vdash J(i), B(i)$. The result follows by composing with the strict curry-uncurry isomorphism. \square

Lemma 4.8 (In \mathbb{C}) Given a pair of morphisms $f: \mathcal{U}_J^I(B, C)$, $g: \mathcal{U}_J^I(A, B)$, if $f \circ g$ is total then so is g .

Proof. A morphism $h: \mathcal{U}_J^I(A, B)$ is total if and only if $\prod_{i:I, a:A(i)} \text{is}_0(h_i a) \rightarrow 0$. For $i: I, a: A(i)$ we have $\text{is}_0(g_i a) \rightarrow \text{is}_0((f \circ g)_i(a))$ by the definition of \circ , and so if $f \circ g$ is total we get $\text{is}_0(g_i a) \rightarrow 0$. \square

Corollary 4.9 (In \mathbb{C}) All isomorphism in \mathcal{U}_J^I are total.

Proof. By induction, totality transfers along paths. Hence, the claim follows since id is total. \square

Proposition 4.10 (In \mathbb{C}) For all types $I: \mathcal{U}$ and families $J, A, B: \mathcal{U}^I$ we have $(A \cong_{\mathcal{U}^I} B) \simeq (A \cong_{\mathcal{U}_J^I} B)$.

Proof. We have a chain of maps $u: \mathcal{U}^I(A, B) \rightarrow \mathcal{U}_{J,\text{tot}}^I(A, B) \rightarrow \mathcal{U}_J^I(A, B)$. The map u strictly preserves identities and composition and therefore lifts to subtypes of isomorphisms (recall that being an isomorphism is a proposition by Corollary 2.7) via $v: (A \cong_{\mathcal{U}^I} B) \rightarrow (A \cong_{\mathcal{U}_J^I} B)$, $\langle f, s, S, r, R \rangle \mapsto \langle uf, us, \text{ap}_u S, ur, \text{ap}_u R \rangle$. Our goal is to show that this restriction is an equivalence. The fibers of u (and thus also v) are propositions, since the first component of u is an equivalence by Corollary 4.7 and the second component is an embedding by Lemma 4.5. Hence, $\text{ap}_u: (f =_{\mathcal{U}^I(A, B)} g) \rightarrow (uf =_{\mathcal{U}_J^I(A, B)} ug)$ is an equivalence for all $f, g: \mathcal{U}^I(A, B)$. The fibers of u are inhabited over isomorphisms and their sections and retractions by Corollary 4.9 and the fact that sections and retractions of isomorphisms are isomorphisms. Thus, the fibers of v are inhabited. \square

4.2 Familial categorical univalence

To verify that $\mathbf{Poly}(\mathbb{C})$ inherits $\text{CUA}_{\mathcal{U}}^\bullet$, we analyze the wild category \mathcal{U}^I in this model. In $\mathbf{Poly}(\mathbb{C})$, we have for $I: \mathcal{U}$ the type \mathcal{U}^I of I -indexed families. Over $A, B: \mathcal{U}^I$, we have the type $A \cong_{\mathcal{U}^I} B$ of isomorphisms between them. We analyze the shapes of these types (i.e. the image under $-_S$) in the base model \mathbb{C} . For clarity, we use different notation: define $I: \mathcal{U} \vdash \mathbf{Fam}(I) := \mathcal{U}^I$ and $I: \mathcal{U}, A, B: \mathbf{Fam}(I) \vdash \mathbf{Iso}(I, A, B) := (A \cong_{\mathcal{U}^I} B)$. Now \mathcal{U}_S is a closed type of \mathbb{C} , \mathbf{Fam}_S is a family of types over it, \mathbf{Iso}_S is a family over \mathcal{U}_S and two copies of \mathbf{Fam}_S , and \mathbf{Iso}_P is a family over \mathbf{Iso}_S .

Remark 4.11 Note that the following data is more ordered than it might seem at first. (1) is exactly the data of an isomorphism in the wild category \mathcal{U}^{I_S} . (2) is the data of an isomorphism in the wild category \mathcal{U}_K^J for some J, K , modulo the first equivalence. Viewing the morphisms in this wild category again as partial functions, the data given by (3) are exactly the inputs on which the functions are not defined.

Lemma 4.12 *Let $I \in \mathbf{Tm}_{\mathbf{Poly}(\mathbb{C})}(\Gamma, \mathcal{U})$ and $A, B \in \mathbf{Tm}_{\mathbf{Poly}(\mathbb{C})}(\Gamma, \mathbf{Fam}(I))$. The shapes of the type $\mathbf{Iso}(I, A, B) \in \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ are equivalent to the iterated Σ type in context Γ_S given by the following:*

$$\begin{aligned} s: \prod_{i:I_S} B_S(i) \longrightarrow A_S(i), \quad f: \prod_{i:I_S} A_S(i) \longrightarrow B_S(i), \quad r: \prod_{i:I_S} B_S(i) \longrightarrow A_S(i), \\ S: f \circ s = \text{id} \quad \text{in } \mathcal{U}^{I_S}(B_S, B_S), \quad R: r \circ f = \text{id} \quad \text{in } \mathcal{U}^{I_S}(A_S, A_S), \end{aligned} \quad (1)$$

as well as the following functions and paths over this equivalence

$$\begin{aligned} \tilde{f}: \prod_{\substack{i:I_S \\ a:A_S(i)}} B_P(f(i, b)) \longrightarrow (1 + I_P(i)) + A_P(i, a), \\ \tilde{s}: \prod_{\substack{i:I_S \\ b:B_S(i)}} A_P(s(i, b)) \longrightarrow (1 + I_P(i)) + B_P(i, b), \quad \tilde{r}: \prod_{\substack{i:I_S \\ a:B_S(i)}} A_P(r(i, b)) \longrightarrow (1 + I_P(i)) + B_P(i, b), \\ S_*(\tilde{s} \circ (\tilde{f}s)) = \text{id} \quad \text{in } \mathcal{U}_{1+I_P}^{\sum_{I_S} B_S}(B_P, B_P), \quad R_*(\tilde{f} \circ (\tilde{r}f)) = \text{id} \quad \text{in } \mathcal{U}_{1+I_P}^{\sum_{I_S} A_S}(A_P, A_P), \end{aligned} \quad (2)$$

where $(\tilde{f}s)(i, b, u) := \tilde{f}(i, s(i, b), u)$ and $(\tilde{r}f)(i, a, u) := \tilde{r}(i, f(i, a), u)$. The family of positions of $A \cong_{\mathcal{U}} B$ is equivalent to the following family over the above characterization of the type of shapes

$$\sum_{\substack{i:I_S, a:A_S(i) \\ u:B_P(a, f(i, a))}} \text{iso}_0(\tilde{f}(i, a, u)) + \sum_{\substack{i:I_S, b:B_S(i) \\ u:A_P(b, s(i, b))}} \text{iso}_0(\tilde{s}(i, b, u)) + \sum_{\substack{i:I_S, b:B_S(i) \\ u:A_P(b, r(i, b))}} \text{iso}_0(\tilde{r}(i, b, u)). \quad (3)$$

Proof. The type $\Gamma \vdash A \cong_{\mathcal{U}} B$ is the Σ type given by $\Gamma \vdash f: \prod_{i:I} A(i) \rightarrow B(i)$, $\Gamma \vdash s, r: \prod_{i:I} B(i) \rightarrow A(i)$, and $\Gamma \vdash fs = \text{id}$, $\Gamma \vdash rf = \text{id}$. By Proposition 3.7, the shape component of a Σ type is the Σ type of the shapes, and the position component of a Σ type is given by the coproduct of the positions. For the families of functions, these are characterized by Lemma 4.1. By associativity of Σ types, and the curry-uncurry isomorphism these correspond to the six families of functions in (1) and (2).

By Proposition 3.8, the shape of an identity type is the identity type of the shapes. As identity types of Σ types, these are equivalent to Σ types of identity types between the first and second component [36, Theorem 9.3.4]. Since identity types respect the strict isomorphism used above up to equivalence, we see that the shape components of the two identity types are equivalent to the four identity types in (1) and (2).

The composition corresponds to composition in the claimed category by Lemma 4.2. By Proposition 3.8, identity types have empty positions yielding together with Lemma 4.1 the above description given in (3). \square

Remark 4.13 We sketch the unfolding of $(A \simeq B) \in \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$. The data given by the functions f, s, r is the same as in Lemma 4.12. The homotopies contribute $r_{SS} \circ f_{SS} \sim \text{id}_{A_S}$ and $f_{SS} \circ s_{SS} \sim \text{id}_{B_S}$ to the shape part. Unlike in Lemma 4.12, however, the homotopies do not encode any relationship between f_{SP}, s_{SP} , and r_{SP} ; a homotopy in $\mathbf{Poly}(\mathbb{C})$ unfolds only to a homotopy between shape components in \mathbb{C} . As such, the homotopies in $(A \simeq B)_S$ witness an equivalence only between the shape components of A and B . The family of positions of $A \simeq B$ agrees with that of $A \cong B$ in Lemma 4.12.

Lemma 4.14 (In \mathbb{C}) *If $\mathbb{C} \vDash \text{CUA}_{\mathcal{U}}^\bullet$, then $\mathbf{Iso}_S(I, A, B) \simeq (\sum_{e:A_S \cong_{\mathcal{U}} B_S} A_P \cong_{\mathcal{U}} \tilde{r} B_P e)$ where $\tilde{I} := \sum_{I_S} A_S$.*

Proof. Let $I \doteq (I_S, I_P): \mathcal{U}_S$, $A \doteq (A_S, A_P), B \doteq (B_S, B_P): \mathbf{Fam}_S(I)$. Set $J(i) := 1 + I_P(i)$ and $\tilde{I} := \sum_{i:I_S} A_S(i)$. Note that the components of $\mathbf{Iso}_S(I, A, B)$ given in Lemma 4.12 (1) are equivalent to $A_S \cong_{\mathcal{U}} B_S$. Denote the remaining components given in Lemma 4.12 (2) by $E(I, A, B)$. It suffices to give for each $e: A_S \cong_{\mathcal{U}} B_S$ an equivalence $E(I, A, B, e) \simeq (A_P \cong_{\mathcal{U}} \tilde{r} B_P e)$. By the fundamental theorem of identity types [36, Theorem 11.2.2] and $\text{CUA}_{\mathcal{U}}^\bullet$, it suffices to consider the case where $A_S \doteq B_S$ and $e \doteq \text{id}$. But in this case $E(I, A, B, \text{id})$ reduces to $A_P \cong_{\mathcal{U}} \tilde{r} B_P$ which is equivalent to $A_P \cong_{\mathcal{U}} \tilde{r} B_P$ by Proposition 4.10. \square

Lemma 4.15 (In \mathbb{C}) *If $\mathbb{C} \vDash \text{CUA}_{\mathcal{U}}^\bullet$, then $\sum_{B:\mathbf{Fam}_S(I)} \mathbf{Iso}_S(I, A, B)$ is contractible for $I: \mathcal{U}_S, A: \mathbf{Fam}_S(I)$.*

Proof. By Lemma 4.14 and $\text{CUA}_{\mathcal{U}}^{\bullet}$ with Lemma 2.10. \square

Lemma 4.16 *A type of $\mathbf{Poly}(\mathbb{C})$ is a proposition exactly if its image under $-_S$ is: naturally in $\Gamma \in \mathbf{Poly}(\mathbb{C})$, given $A \in \text{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ there is a logical equivalence $\text{Tm}_{\mathbb{C}}(\Gamma_S, \text{is-prop}(A_S)) \longleftrightarrow \text{Tm}_{\mathbf{Poly}(\mathbb{C})}(\Gamma, \text{is-prop}(A))$.*

Proof. By Proposition 3.8, we have that identity types are in the image of \circ_S and preserved by $-_S$. Thus, $\text{Tm}(\Gamma.A.A, \text{qp} =_A \text{q}) \cong \text{Tm}(\Gamma_S.A_S.A_S, (\text{qp} =_A \text{q})_S) \cong \text{Tm}(\Gamma_S.A_S.A_S, \text{qp} =_{A_S} \text{q})$. \square

Theorem 4.17 *If $\mathbb{C} \models \text{CUA}_{\mathcal{U}}^{\bullet}$, then $\mathbf{Poly}(\mathbb{C}) \models \text{CUA}_{\mathcal{U}}^{\bullet}$.*

Proof. By Lemma 2.10, it suffices to show $I: \mathcal{U}, A: \text{Fam}(I) \vdash \sum_{B: \text{Fam}(I)} \text{Iso}(I, A, B)$ is contractible. It is inhabited by the identity, so it is enough to show it is a proposition. By Lemma 4.16, it suffices to show $I: \mathcal{U}_S, A: \text{Fam}(I)_S \vdash \sum_{B: \text{Fam}_S(I)} \text{Iso}_S(I, A, B)$ is a proposition in \mathbb{C} , and this is Lemma 4.15. \square

Remark 4.18 Von Glehn [50, §5.1] observes that the outputs of $\mathbf{Poly}(-)$ are also suitable inputs to $\mathbf{Poly}(-)$, meaning the construction can be iterated. Theorem 4.17 implies that iterated polynomial models also inherit $\text{CUA}_{\mathcal{U}}^{\bullet}$ from the base model, though we do not know if there is any use for these models.

5 Familial categorical univalence without function extensionality

Using the results from Section 4 together with Von Glehn's counterexample to function extensionality in $\mathbf{Poly}(\mathbb{C})$, which we recall in Section 5.1, we can derive the independence of $\text{FE}_{\mathcal{U}}$ from $\text{ITT} + \text{CCUA}_{\mathcal{U}}$.

5.1 Failure of function extensionality in the polynomial model

Von Glehn's proof that FE fails in $\mathbf{Poly}(\mathbb{C})$ [50, Proposition 4.11] uses the following types:

Definition 5.1 Given $\Gamma \in \mathbf{Poly}(\mathbb{C})$ and $A \in \text{Ty}_{\mathbb{C}}(\Gamma_S)$, define $\top\langle A \rangle \in \text{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ by $\Gamma_S \vdash \top\langle A \rangle_S := 1$ and $\Gamma_S.\top\langle A \rangle_S \vdash \top\langle A \rangle_P := A$.

Proposition 5.2 *There exist $f_0, f_1 \in \text{Tm}(1, \top\langle 1+1 \rangle \rightarrow \top\langle 1 \rangle)$ with $(f_0 = f_1) \rightarrow 0$.*

Proof. For each $k \in \{0, 1\}$, we define $b_k \in \text{Tm}(1.\top\langle 1+1 \rangle, \top\langle 1 \rangle)$ by setting $1.1 \vdash (b_k)_S := \star: 1$ and $1.1.1 \vdash (b_k)_P := \text{in}_k(\star): 1+1$. Take $f_k := \lambda(b_k)$. Unfolding the construction of λ in Proposition 3.12, we have $(f_k)_S \doteq \langle (b_k)_S, (b_k)_P \rangle$, so $(f_0 = f_1)_S$ implies $(b_0)_P = (b_1)_P$ and is thus empty. \square

Proposition 5.3 ([50, Proposition 4.11]) $\mathbf{Poly}(\mathbb{C}) \models \neg \text{FE}_{\mathcal{U}}$.

Proof. The functions from Proposition 5.2 are homotopic since the codomain is a proposition by Lemma 4.16. Note that $\top\langle A \rangle$ belongs to the universe of $\mathbf{Poly}(\mathbb{C})$ for $A: \mathcal{U}$. \square

5.2 Independence of function extensionality from familial categorical univalence

Proposition 5.4 *There is a model \mathbb{C} of ITT with extensive finite coproducts satisfying the strict η rule such that $\mathbb{C} \models \text{FE}$ and $\mathbb{C} \models \text{UA}_{\mathcal{U}}$.*

Proof. Take the model of $\text{ITT} + \text{FE} + \text{UA}_{\mathcal{U}}$ constructed by Cohen, Coquand, Huber, and Mörtberg [15], whose category of contexts is the category of presheaves on the De Morgan cube category and whose types are dependent cubical sets equipped with a uniform Kan filling operation. Orton and Pitts [34, Theorem 5.14] show that binary coproducts of types can be modeled by coproducts of dependent cubical sets, and it is easy to check the same for nullary coproducts. Thus these coproducts satisfy the strict η law and, since every topos is an extensive category [12, Remark 4.10], are also extensive. \square

Remark 5.5 The particular choice of cubical model is not important in the proof above; any model in the style of Orton and Pitts [34] or Angiuli et al. [4] will do, as will Voevodsky's (non-constructive) simplicial model [29]. There are, however, models of $\text{ITT} + \text{UA}_{\mathcal{U}}$ that do not support extensive finite coproducts of types; see for example the need for a factorization in Shulman [39, Proposition 6.2].

Theorem 5.6 $\text{ITT} + \text{CUA}_{\mathcal{U}}^{\bullet} \not\vdash \text{FE}_{\mathcal{U}}$.

Proof. Take \mathbb{C} to be a model of $\text{ITT} + \text{FE} + \text{UA}_{\mathcal{U}}$ with extensive finite coproducts of types, as provided by Proposition 5.4. The combination $\text{FE} + \text{UA}_{\mathcal{U}}$ implies $\text{CUA}_{\mathcal{U}}^{\bullet}$, as FE tells us that $(A =_{I \rightarrow \mathcal{U}} B) \simeq (A \sim B)$. Thus $\text{Poly}(\mathbb{C}) \vDash \text{ITT} + \text{CUA}_{\mathcal{U}}^{\bullet}$ by Theorem 4.17, while $\text{Poly}(\mathbb{C}) \not\vdash \text{FE}_{\mathcal{U}}$ by Proposition 5.3. \square

6 Variations

Once $\text{UA}_{\mathcal{U}}$ was proposed by Voevodsky, it was quickly taken up as the canonical axiom for its intended purpose. Inequivalent variations on $\text{UA}_{\mathcal{U}}$ usually turn out to be significantly weaker, as in case of the “isomorphism reflection” that holds in Bauer and Winterhalter’s cardinal model [51, §8.3], or else inconsistent, as in the case of “qinv-univalence” [44, Exercise 4.6].

Unfortunately, we do not see evidence for a canonical form of “FE-free univalence”. In this section, we show that a few possible candidates are inequivalent; none stands out as the most natural. In Section 6.1, we show that $\text{CUA}_{\mathcal{U}}$ does not imply $\text{CUA}_{\mathcal{U}}^{\bullet}$. In Section 6.2, we identify an axiom $\text{CCUA}_{\mathcal{U}}$ that also satisfies $\text{ITT} + \text{CCUA}_{\mathcal{U}} \not\vdash \text{FE}_{\mathcal{U}}$ and $\text{ITT} + \text{CCUA}_{\mathcal{U}} + \text{FE}_{\mathcal{U}} \vdash \text{UA}_{\mathcal{U}}$ but is not equivalent to $\text{CUA}_{\mathcal{U}}$ or $\text{CUA}_{\mathcal{U}}^{\bullet}$.

In Section 6.3, we recall a variant of univalence used by Van den Berg [46, Definition 2.13] which we call *approximate univalence* or $\text{UA}_{\mathcal{U}}^{\sim}$. It is an open question whether $\text{UA}_{\mathcal{U}}^{\sim}$ implies $\text{FE}_{\mathcal{U}}$; we do not resolve the question, but we pose a related question that avoids mention of a universe.

6.1 Non-familial categorical univalence

Our Theorem 5.6 is a priori more than an answer to Dorais’ question of whether $\text{ITT} + \text{CUA}_{\mathcal{U}}$ proves $\text{FE}_{\mathcal{U}}$: we prove not only that $\text{ITT} + \text{CUA}_{\mathcal{U}} \not\vdash \text{FE}_{\mathcal{U}}$ but that $\text{ITT} + \text{CUA}_{\mathcal{U}}^{\bullet} \not\vdash \text{FE}_{\mathcal{U}}$. One may then wonder if $\text{CUA}_{\mathcal{U}}^{\bullet}$ is strictly stronger than $\text{CUA}_{\mathcal{U}}$. This is indeed the case.

Theorem 6.1 $\text{ITT} + \text{CUA}_{\mathcal{U}} \not\vdash \text{CUA}_{\mathcal{U}}^{\bullet}$.

Proof. Take \mathbb{C} to be a model of $\text{ITT} + \text{FE} + \text{UA}_{\mathcal{U}}$ with extensive finite coproducts of types, as provided by Proposition 5.4. Then $\text{Poly}(\mathbb{C}) \vDash \text{CUA}_{\mathcal{U}}^{\bullet}$ by Theorem 4.17, and in particular $\text{Poly}(\mathbb{C}) \vDash \text{CUA}_{\mathcal{U}}$. We now consider the slice model $\text{Poly}(\mathbb{C})/\mathbb{T}\langle 1 \rangle$ for $\mathbb{T}\langle - \rangle$ from Definition 5.1. That is, we work in the context $t: \mathbb{T}\langle 1 \rangle$. However, we modify the interpretation of the universe \mathcal{U} .

Define $(\mathcal{U}', \mathbf{El}')$ by $\mathcal{U}' := \mathcal{U} \times \mathbb{T}\langle 1 \rangle \in \text{Ty}_{\text{Poly}(\mathbb{C})}(1)$ with $\mathbf{El}'\langle A, t \rangle := A \in \text{Ty}_{\text{Poly}(\mathbb{C})}(\mathcal{U}')$. In the slice model, $\mathbb{T}\langle 1 \rangle$ is contractible by Lemma 4.16, so this universe is closed under the same type formers as \mathcal{U} and the projection $\pi: \mathcal{U}' \rightarrow \mathcal{U}$ is an equivalence. For $A, B: \mathcal{U}'$ in the slice model, the interpretation of id-to-ceq for \mathcal{U}' is homotopic (by path induction) to the composite of $\text{ap}_{\pi}: (A =_{\mathcal{U}'} B) \rightarrow (\pi A =_{\mathcal{U}} \pi B)$ followed by $\text{id-to-ceq}: (\pi A =_{\mathcal{U}} \pi B) \rightarrow (\pi A \cong \pi B)$. Since ap_{π} is an equivalence, $\text{CUA}_{\mathcal{U}'}$ holds in the slice model.

To see that $\text{CUA}_{\mathcal{U}'}$ fails in the slice model, take $I := \mathbb{T}\langle 1 + 1 \rangle$ and recall from Proposition 5.2 that there exist distinct $f_0 \neq f_1: I \rightarrow \mathbb{T}\langle 1 \rangle$. For $k \in \{0, 1\}$, set $A_k := (\lambda i. \langle 1, f_k(i) \rangle) \in (\mathcal{U}')^I$. Then $A_0 \cong_{(\mathcal{U}')^I} A_1$ is by definition $1 \cong_{(\mathcal{U}')^I} 1$ and thus inhabited, while $A_0 =_{(\mathcal{U}')^I} A_1$ would imply $f_0 = f_1$ and is thus empty. \square

6.2 Categorical categorical univalence

In our formulation of $\text{CUA}_{\mathcal{U}}$, we could have required that id-to-ceq be a *categorical* equivalence.

Definition 6.2 *Categorical categorical univalence* ($\text{CCUA}_{\mathcal{U}}$) is the principle that the canonical map $\text{id-to-ceq}: (A =_{\mathcal{U}} B) \rightarrow (A \cong B)$ is a categorical equivalence for all $A, B: \mathcal{U}$.

A point in favor of $\text{CCUA}_{\mathcal{U}}$ is that it is a proposition (cf. Corollary 2.7); the structure of “being an equivalence” need not be a proposition without FE (cf. implication (iii) \implies (viii) of Theorem 2.13). However, it is unusually strong relative to other identity type characterizations. For example, the equivalence $(\langle a, b \rangle =_{A \times B} \langle a', b' \rangle) \simeq (a =_A a') \times (b =_B b')$ characterizing identities in Σ types cannot be shown to be a categorical equivalence in ITT . $\text{CCUA}_{\mathcal{U}}$ also seems brittle. Note that the “canonical” map $\text{id-to-ceq}: A =_{\mathcal{U}} B \rightarrow A \cong B$ is only canonically defined *up to homotopy* by the requirement $\text{id-to-ceq}(\text{refl}_A) = \text{id}$! It is not clear to us that different formulations of $\text{CCUA}_{\mathcal{U}}$ using homotopic definitions of id-to-ceq are interderivable.

In any case, using $\mathbf{Poly}(-)$, we will show that $\mathbf{CCUA}_{\mathcal{U}}$ is strictly stronger than $\mathbf{CUA}_{\mathcal{U}}$ and moreover not implied by $\mathbf{CUA}_{\mathcal{U}}^{\bullet}$, yet still does not imply $\mathbf{FE}_{\mathcal{U}}$. We strengthen Theorem 4.17 to $\mathbf{CCUA}_{\mathcal{U}}$ by exploiting properties of types in the essential image of \circ_S . By general properties of reflective subcategories, these are exactly those with strictly invertible unit. In fact, they are also exactly those with categorically invertible unit, but we will not need this.

Proposition 6.3 *Naturally in Γ , for $A \in \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$, the following are equivalent:*

- (i) $\eta_A: A \rightarrow \circ_S A$ is a strict isomorphism,
- (ii) there is a map $A_P \rightarrow 0$ in $\mathbf{Ty}_{\mathbb{C}}(\Gamma_S.A_S)$,
- (iii) A_P is strictly isomorphic to 0 in $\mathbf{Ty}_{\mathbb{C}}(\Gamma_S.A_S)$.

Proof. Consider the morphism $\eta_A: \Gamma.A \rightarrow \Gamma.\circ_S A$ in $\mathbf{Poly}(\mathbb{C})$ over Γ . The shape component is given by $\text{id}_A: \Gamma_S.A_S \rightarrow \Gamma_S.A_S$. The positions component is given by $[\text{in}_0, !_{A_P}]: \Gamma_S.A_S.\Gamma_P \mathbf{p} + 0 \rightarrow \Gamma_P.A_S.\Gamma_P \mathbf{p} + A_P$. Since the shape component is an isomorphism, η_A is an isomorphism exactly if the position component is.

We work in the internal language of \mathbb{C} . The direction (iii) \implies (i) is clear. The equivalence (ii) \iff (iii) follows from the strict η rule for 0. It is left to show (i) \implies (ii). Suppose we are given a family of strict inverses $\lambda a.[\text{in}_0, i_a]: \prod_{a:A_S} \Gamma_P + A_P(a) \rightarrow \Gamma_P + 0$ to $\lambda a.[\text{in}_0, \text{elim}_0]: \prod_{a:A_S} \Gamma_P + 0 \rightarrow \Gamma_P + A_P(a)$. Then $\lambda a.i_a: \prod_{a:A_S} A_P(a) \rightarrow \Gamma_P + 0$ and $\lambda a.\text{elim}_0: \prod_{a:A_S} 0 \rightarrow \Gamma_P + A_P(a)$ form an equivalence in $\mathcal{U}_{\Gamma_P}^{A_S}$. Hence, the family of maps i is total by Corollary 4.9. \square

Definition 6.4 A type $A \in \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ is \circ_S -modal if the conditions from Proposition 6.3 hold.

Let $A, B \in \mathbf{Ty}_{\mathbb{C}}(\Gamma)$, $f \in \mathbf{Tm}_{\mathbb{C}}(\Gamma, A \rightarrow B)$, and $F: \mathbb{C} \rightarrow \mathbb{D}$ a pseudomorphism. We write

$$\tilde{F}: \mathbf{Tm}(\Gamma, A \rightarrow B) \longrightarrow \mathbf{Tm}(F\Gamma, FA \rightarrow FB), \quad f \longmapsto \lambda(F(\text{app}(f))).$$

The image of f under $F: \mathbb{C} \rightarrow \mathbb{D}$ is $Ff \in \mathbf{Tm}(F\Gamma, F(A \rightarrow B))$. There is always a comparison map $\lambda(F(\text{app}(q_{A \rightarrow B}))) : F(B^A) \rightarrow (FB)^{(FA)}$ [28, §4], and the image of Ff under this map coincides with $\tilde{F}f$.

Lemma 6.5 *The pseudomorphism $\circ_S: \mathbb{C} \rightarrow \mathbf{Poly}(\mathbb{C})$ preserves paths between functions: naturally in Γ , given $A, B \in \mathbf{Ty}(\Gamma)$ and $f, g \in \mathbf{Tm}(\Gamma, A \rightarrow B)$, there is a map*

$$\mathbf{Tm}(\Gamma, f =_{A \rightarrow B} g) \longrightarrow \mathbf{Tm}(\circ_S \Gamma, \widetilde{\circ_S} f =_{\circ_S A \rightarrow \circ_S B} \widetilde{\circ_S} g).$$

Proof. Let $H \in \mathbf{Tm}(\Gamma, f = g)$. We have $\circ_S H \in \mathbf{Tm}(\circ_S \Gamma, \circ_S(f =_{A \rightarrow B} g))$. By the definition of the identity types in $\mathbf{Poly}(\mathbb{C})$ (Proposition 3.8), they are preserved by the pseudomorphism \circ_S . In particular, we have $\mathbf{Tm}(\circ_S \Gamma, \circ_S(f =_{A \rightarrow B} g)) \cong \mathbf{Tm}(\circ_S \Gamma, \circ_S f =_{\circ_S(A \rightarrow B)} \circ_S g)$. By lifting the comparison map $\circ_S(B^A) \rightarrow (\circ_S B)^{(\circ_S A)}$ to identity types, we obtain the desired element. \square

Corollary 6.6 *The pseudomorphism $\circ_S: \mathbb{C} \rightarrow \mathbf{Poly}(\mathbb{C})$ preserves categorical equivalences: naturally in Γ , given $A, B \in \mathbf{Ty}(\Gamma)$, $f \in \mathbf{Tm}(\Gamma, A \rightarrow B)$ we have a map*

$$\mathbf{Tm}(\Gamma, \text{is-ceq}(f)) \longrightarrow \mathbf{Tm}(\circ_S \Gamma, \text{is-ceq}(\widetilde{\circ_S} f)).$$

Proof. The action on functions $\widetilde{\circ_S}$ preserves composition and identities. \square

The \circ_S -modal types in $\mathbf{Poly}(\mathbb{C})$ behave like types of the base model. In particular, when the base model enjoys function extensionality, homotopy equivalences between \circ_S -modal types can be improved to categorical equivalences.

Lemma 6.7 *If $\mathbb{C} \models \mathbf{FE}$, then equivalences coincide with categorical equivalences between \circ_S -modal types.*

Proof. Let $A, B \in \mathbf{Ty}_{\mathbf{Poly}(\mathbb{C})}(\Gamma)$ and $f: A \rightarrow B$ an equivalence. Since $-_S$ preserves identity types, the map $f_S: A_S \rightarrow B_S$ is an equivalence, and by \mathbf{FE} also a categorical equivalence. The pseudomorphism \circ_S preserves categorical equivalences by Corollary 6.6. Hence, so does the dependent right adjoint \circ_S since

it is defined by substituting the image of the pseudomorphism along the unit. Since $\circ_S f \circ \eta_A \doteq \eta_B \circ f$ the claim follows by Proposition 6.3 and 2-out-of-3 for categorical equivalences (Lemma 2.8). \square

We show that the type family $I: \mathcal{U}, A, B: \mathbf{Fam}(I) \vdash \mathbf{Iso}(I, A, B) := (A \cong_{\mathcal{U}} B)$ from Section 4.2 is \circ_S -modal, which we can use to improve the equivalence in the definition of $\mathbf{CUA}_{\mathcal{U}}^{\bullet}$ to a categorical equivalence over well-behaved base models.

Lemma 6.8 *If $\mathbb{C} \vDash \mathbf{CUA}_{\mathcal{U}}^{\bullet}$, then \mathbf{Iso} in $\mathbf{Poly}(\mathbb{C})$ is \circ_S -modal.*

Proof. By Proposition 6.3, it suffices to show that the positions of $\mathbf{Iso} \in \mathbf{Ty}(1.\mathcal{U}.\mathbf{Fam} \times \mathbf{Fam}, \mathbf{Iso})$ are empty. We work internally to \mathbb{C} . We show for all $I: \mathcal{U}_S, A \doteq (A_S, A_P), B \doteq (B_S, B_P): \mathbf{Fam}_S(I), e: \mathbf{Iso}_S(I, A, B)$ that $\mathbf{Iso}_P(I, A, B, e) \rightarrow 0$, which suffices by strict initiality of 0. By Lemma 4.15, we can assume $A \doteq B$ and $e \doteq \text{id}$. In this case, the functions on positions are given by in_1 and therefore the type (3) is empty. \square

Proposition 6.9 *If \mathbb{C} is a model of $\text{ITT} + \text{FE} + \text{UA}_{\mathcal{U}}$ with extensive finite coproducts of types, then $\mathbf{Poly}(\mathbb{C}) \vDash \mathbf{CCUA}_{\mathcal{U}}$.*

Proof. By Theorem 4.17, we have $\mathbf{Poly}(\mathbb{C}) \vDash \mathbf{CUA}_{\mathcal{U}}$. Since the types declared equivalent in the statement of $\mathbf{CUA}_{\mathcal{U}}$ are \circ_S -modal, by Lemma 6.8 and the definition of identity types (Proposition 3.8), the claim follows from Lemma 6.7. \square

Corollary 6.10 $\text{ITT} + \mathbf{CCUA}_{\mathcal{U}} \not\vdash \text{FE}_{\mathcal{U}}$.

Proof. Proposition 5.4 provides a model \mathbb{C} of $\text{ITT} + \text{FE} + \text{UA}_{\mathcal{U}}$ with extensive finite coproducts of types. We have $\mathbf{Poly}(\mathbb{C}) \vDash \mathbf{CCUA}_{\mathcal{U}}$ by Proposition 6.9 and $\mathbf{Poly}(\mathbb{C}) \vDash \neg \text{FE}_{\mathcal{U}}$ by Proposition 5.3. \square

Finally, we show that while $\mathbf{CCUA}_{\mathcal{U}}$ is still strictly weaker than $\text{UA}_{\mathcal{U}}$, it is strictly stronger than $\mathbf{CUA}_{\mathcal{U}}$.

Proposition 6.11 $\text{ITT} + \mathbf{CUA}_{\mathcal{U}}^{\bullet} \not\vdash \mathbf{CCUA}_{\mathcal{U}}$.

Proof. Take \mathbb{C} to be a model of $\text{ITT} + \text{FE} + \text{UA}_{\mathcal{U}}$ with extensive finite coproducts of types, as provided by Proposition 5.4. Then $\mathbf{Poly}(\mathbb{C}) \vDash \mathbf{CUA}_{\mathcal{U}}^{\bullet}$ by Theorem 4.17. As in the proof of Theorem 6.1, we now consider the slice model $\mathbf{Poly}(\mathbb{C})/\top\langle 1 \rangle$ for $\top\langle - \rangle$ from Definition 5.1. This time, however, we modify the interpretation of the identity type.

We define our new identity types by $(u ='_A v) := (u =_A v) \times \top\langle 1 \rangle$, where $u =_A v$ is the identity type in $\mathbf{Poly}(\mathbb{C})$. Since $\top\langle 1 \rangle$ is a proposition by Lemma 4.16, it is contractible in the slice, so the projection $(u ='_A v) \rightarrow (u =_A v)$ is an equivalence and in particular $u ='_A v$ is an identity type. If we write \cong' for wild-categorical isomorphisms defined with $='$, it follows also that $(a \cong_{\mathbb{D}} b) \simeq (a \cong'_{\mathbb{D}} b)$ for any wild category \mathbb{D} . Thus $\mathbf{CUA}_{\mathcal{U}}^{\bullet}$, which holds in $\mathbf{Poly}(\mathbb{C})$ by Theorem 4.17, transfers to the slice model with the new identity type.

However, $\mathbf{CCUA}_{\mathcal{U}}$ cannot hold (in or out of the slice) when formulated with $='$ and \cong' . For $A, B: \mathcal{U}$, the family of positions for $(A ='_U B)$ is the constant family 1. The family of positions for $A \cong'_U B$, which is categorically equivalent to $(A \cong_U B) \times \top\langle 1 \rangle \times \top\langle 1 \rangle$, is the constant family $1 + 1$ (using Lemma 6.8). Thus, by Lemma 4.14, the equivalence $(A ='_U B) \simeq (A \cong'_U B)$ is only categorical when both sides are empty. \square

6.3 Approximate univalence

Van den Berg [46, Definition 2.13] defines another weak form of $\text{UA}_{\mathcal{U}}$, in the language of path categories, which can be rendered in type theory as follows.

Definition 6.12 *Approximate univalence* ($\text{UA}_{\mathcal{U}}^{\sim}$) is the principle that for all $A, B: \mathcal{U}$ and $e: A \simeq B$, we have some $p: A =_{\mathcal{U}} B$ such that $\text{id-to-eq}(p) \sim e$.

Notably, $\text{UA}_{\mathcal{U}}^{\sim}$ can be expressed as an inference rule without Π types. In the presence of Π types, Swan [43, Remark 4.6] comments that it is an open question whether $\text{UA}_{\mathcal{U}}^{\sim}$ implies $\text{FE}_{\mathcal{U}}$. An immediate but subtle consequence of $\text{UA}_{\mathcal{U}}^{\sim}$ is that there is a composite map $(A \simeq B) \rightarrow (A =_{\mathcal{U}} B) \rightarrow (A \cong B)$ that improves any homotopy equivalence to a homotopic categorical equivalence. In light of the decomposition of $\text{UA}_{\mathcal{U}}$ in Section 2.2, it is natural to consider an analogue of Definition 2.11:

Definition 6.13 *Approximate equivalence improvement* (EI^\sim) is the principle that for all types A, B and $e: A \simeq B$, we have some $e': A \cong B$ such that $\text{ceq-to-eq}(e') \sim e$.

One FE-like corollary of EI^\sim is that if P is a contractible type, then $A \rightarrow P$ is also contractible for every type A : by Lemma 2.5, we have $(A \rightarrow P) \simeq (A \rightarrow 1) \cong 1$. This is not provable in ITT, as Lemma 4.16 and Proposition 5.2 show. However, the exact relationship between EI^\sim and FE is a mystery to us:

Question 6.14 *Does $\text{ITT} + \text{EI}^\sim \vdash \text{FE}$?*

An answer to Question 6.14 might not tell us whether $\text{ITT} + \text{UA}_{\mathcal{U}}^\sim \vdash \text{FE}_{\mathcal{U}}$, but it may be a more tractable question. The polynomial models refute EI^\sim : $\mathbb{T}\langle 1 \rangle$ and $\mathbb{T}\langle 1 + 1 \rangle$ are equivalent (Remark 4.13) but not categorically equivalent (Lemma 4.14). Boulier, Pédrot, and Tabareau’s *intensional function translation* [10, §3] sends a theory with FE to a syntactic model with $\text{EI}^\sim \wedge \neg \text{FE}$, but its function types do not satisfy any η rule, so this does not answer the question for ITT as we define it. Shulman [38] has a recipe for expressing universal properties without FE that suggests stronger forms of EI^\sim ; for example, one can also ask that homotopic categorical equivalences are equal. It is not clear to us how these strengthenings relate to EI^\sim or to FE.

Remark 6.15 Naturally, we can also consider *approximate categorical univalence* $\text{CUA}_{\mathcal{U}}^\sim$: the principle that for all $A, B: \mathcal{U}$ and $e: A \cong B$, we have some $p: A =_{\mathcal{U}} B$ such that $\text{id-to-ceq}(p) \sim e$. This is the weakest of all the univalence principles we have considered, but we do not know if it is strictly weaker than $\text{CUA}_{\mathcal{U}}$.

7 Related work

To conclude, we comment on the status of weak forms of univalence in other known models of type theory without function extensionality.

7.1 Realizability models

Realizability is a standard source of models of ITT that refute extensionality principles, including FE; see Streicher [42, Theorem 2.9, §3.7]. However, most work combining features of realizability and homotopical semantics, such as that of Frumin and Van den Berg [20] and Uemura [45], constructs models that *do* satisfy FE. An exception is Speight’s *groupoidal realizability* [41]; his function types have neither FE nor the η rule. Speight constructs an impredicative universe of modest fibrations, but we do not know if this or any other universe in the model satisfies some kind of univalence.

7.2 Pédrot and Tabareau’s parametric exceptional translation

The *parametric exceptional translation* [35] is another source of models of type theory without FE. Presented as a syntactic translation, it induces a construction $\mathbf{ParEx}(-)$ on models. Unlike $\mathbf{Poly}(-)$, however, $\mathbf{ParEx}(-)$ does not preserve any form of univalence that we know of. We sketch here a reason for the simplest form of the translation ($\mathbb{E} = 1$ and $\Omega_i(\star) = 1$). Kovács [31] has formalized this case in Agda.

Given a model \mathbb{C} , the category of contexts in $\mathbf{ParEx}(\mathbb{C})$ is $\int_{\Gamma \in \mathbb{C}} \mathbf{Ty}(\Gamma)$: objects are pairs $\Gamma = (\Gamma_S, \Gamma_P)$ as in $\mathbf{Poly}(\mathbb{C})$, but a morphism $\sigma: \Delta \rightarrow \Gamma$ is a pair of $\sigma_S: \Delta_S \rightarrow \Gamma_S$ in \mathbb{C} and $\sigma_P: \Delta_P \rightarrow \Gamma_P \sigma_S$ in $\mathbf{Ty}(\Delta_S)$. We think of Γ_P as selecting “valid” elements of Γ_S . Types $A \in \mathbf{Ty}(\Gamma)$ have components $A_S \in \mathbf{Ty}(\Gamma_S)$, $A_P \in \mathbf{Ty}(\Gamma_S.\Gamma_P.A_S)$, and $A_E \in \mathbf{Tm}(\Gamma, A_S)$. Again we think of A_P as selecting valid elements of A_S , while A_E is a distinguished “error” element. While intuitively A_E should not be valid, this is not enforced.

The mismatch between $A \cong B$ and $A =_{\mathcal{U}} B$ is that for the former, categorical equivalences of the $-_S$ and $-_P$ components suffice, while the latter requires also that A_E corresponds to B_E . For example, define $X^0, X^1 \in \mathbf{Ty}(1)$ by $X_S^k := 1 + 1$, $X_P^k(b) := 1$, and $X_E^k = \text{in}_k(\star)$. The identity equivalence on $1 + 1$ defines a strict isomorphism $X^0 \cong X^1$ that cannot induce a path $X^0 =_{\mathcal{U}} X^1$ because it does not send X_E^0 to X_E^1 .

7.3 Bordg’s projective model

Bordg [8,9] describes a model of type theory with Σ types, Π types, identity types, and a universe \mathcal{U} in the category $[\mathbf{BC}_2, \mathbf{Gpd}]$ of groupoid-valued presheaves on the two-element group. This model is based

on the *projective Quillen model structure*: types are morphisms of $[\mathbf{BC}_2, \mathbf{Gpd}]$ whose underlying \mathbf{Gpd} -morphisms are isofibrations. Bordg observes that the model refutes FE [9, Proposition 6.3] and that the natural choice of universe is not univalent [9, Proposition 6.1], despite the fact that a stronger condition on types corresponding to the *injective model structure* yields a model of FE and $\mathbf{UA}_{\mathcal{U}}$ [8, §5.4].

Considering our weaker forms of $\mathbf{UA}_{\mathcal{U}}$ in this model is fruitless: although FE fails, $\mathbf{FE}_{\mathcal{U}}$ holds (cf. [9, Remark 6.4]). This is because the groupoids classified by \mathcal{U} are strict sets, as in Hofmann and Streicher’s groupoid model [26]. Thus, $\mathbf{CUA}_{\mathcal{U}}$ for example cannot hold, for if it did then $\mathbf{UA}_{\mathcal{U}}$ would follow.

References

- [1] Abbott, M., T. Altenkirch and N. Ghani, *Categories of containers*, in: A. D. Gordon, editor, *Foundations of Software Science and Computation Structures (FoSSaCS 2003)*, volume 2620 of *Lecture Notes in Computer Science*, pages 23–38, Springer Berlin Heidelberg (2003).
https://doi.org/10.1007/3-540-36576-1_2
- [2] Abbott, M. G., *Categories of Containers*, Ph.D. thesis, University of Leicester (2003).
<https://hdl.handle.net/2381/30102>
- [3] Altenkirch, T. and A. Kaposi, *A container model of type theory* (2021). Abstract for a presentation at TYPES 2021.
<https://types21.liacs.nl/download/a-container-model-of-type-theory>
- [4] Angiuli, C., G. Brunerie, T. Coquand, R. Harper, K.-B. Hou (Favonia) and D. R. Licata, *Syntax and models of cartesian cubical type theory*, *Mathematical Structures in Computer Science* **31**, pages 424–468 (2021).
<https://doi.org/10.1017/S0960129521000347>
- [5] Angiuli, C. and D. Gratzer, *Principles of Dependent Type Theory* (2025). Draft of a book to be published by Cambridge University Press.
<https://www.danielgratzer.com/papers/type-theory-book.pdf>
- [6] Awodey, S., *Natural models of homotopy type theory*, *Mathematical Structures in Computer Science* **28**, pages 241–286 (2016).
<https://doi.org/10.1017/s0960129516000268>
- [7] Bezem, M., T. Coquand and S. Huber, *A model of type theory in cubical sets*, in: R. Matthes and A. Schubert, editors, *19th International Conference on Types for Proofs and Programs, TYPES 2013, Toulouse, France, April 22-26, 2013*, volume 26 of *LIPICs*, pages 107–128, Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2014).
<https://doi.org/10.4230/LIPICs.TYPES.2013.107>
- [8] Bordg, A., *On lifting univalence to the equivariant setting*, Ph.D. thesis, Université Nice Sophia Antipolis (2015).
<https://arxiv.org/abs/1512.04083>
- [9] Bordg, A., *On the inadequacy of the projective structure with respect to the univalence axiom* (2020). Preprint.
<https://arxiv.org/abs/1712.02652>
- [10] Boulier, S., P.-M. Pédrot and N. Tabareau, *The next 700 syntactical models of type theory*, in: *Proceedings of the 6th ACM SIGPLAN Conference on Certified Programs and Proofs (CPP 2017)*, pages 182–194, ACM (2017).
<https://doi.org/10.1145/3018610.3018620>
- [11] Capriotti, P. and N. Kraus, *Univalent higher categories via complete semi-Segal types*, *Proceedings of the ACM on Programming Languages* **2**, pages 1–29 (2017).
<https://doi.org/10.1145/3158132>
- [12] Carboni, A., S. Lack and R. Walters, *Introduction to extensive and distributive categories*, *Journal of Pure and Applied Algebra* **84**, pages 145–158 (1993).
[https://doi.org/10.1016/0022-4049\(93\)90035-r](https://doi.org/10.1016/0022-4049(93)90035-r)
- [13] Cartmell, J. W., *Generalised Algebraic Theories and Contextual Categories*, Ph.D. thesis, Oxford University (1978).
- [14] Clairambault, P. and P. Dybjer, *The biequivalence of locally cartesian closed categories and Martin-Löf type theories*, *Mathematical Structures in Computer Science* **24**, page e240606 (2014).
<https://doi.org/10.1017/S0960129513000881>
- [15] Cohen, C., T. Coquand, S. Huber and A. Mörtberg, *Cubical type theory: A constructive interpretation of the univalence axiom*, in: T. Uustalu, editor, *21st International Conference on Types for Proofs and Programs (TYPES 2015)*, volume 69 of *Leibniz International Proceedings in Informatics (LIPICs)*, pages 5:1–5:34, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2018).
<https://doi.org/10.4230/LIPICs.TYPES.2015.5>

- [16] Coquand, T., F. Ruch and C. Sattler, *Constructive sheaf models of type theory*, *Mathematical Structures in Computer Science* **31**, pages 979–1002 (2021).
<https://doi.org/10.1017/S0960129521000359>
- [17] de Paiva, V., *The Dialectica Categories*, Ph.D. thesis, University of Cambridge (1991).
<https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-213.pdf>
- [18] Dorais, F. G., *Equivalent form of the univalence axiom*, *MathOverflow* (2013). Version: 2013-06-22.
<https://mathoverflow.net/q/134449>
- [19] Dybjer, P., *Internal type theory*, in: *Types for Proofs and Programs: International Workshop TYPES '95, Torino, Italy*, pages 120–134, Springer Berlin Heidelberg (1996).
https://doi.org/10.1007/3-540-61780-9_66
- [20] Frumin, D. and B. van den Berg, *A homotopy-theoretic model of function extensionality in the effective topos*, *Mathematical Structures in Computer Science* **29**, pages 588–614 (2018), ISSN 1469-8072.
<https://doi.org/10.1017/s0960129518000142>
- [21] Gambino, N. and J. Kock, *Polynomial functors and polynomial monads*, *Mathematical Proceedings of the Cambridge Philosophical Society* **154**, pages 153–192 (2013).
<https://doi.org/10.1017/s0305004112000394>
- [22] Ghani, N., *$\beta\eta$ -Equality for coproducts*, pages 171–185, *Lecture Notes in Computer Science*, Springer Berlin Heidelberg (1995).
<https://doi.org/10.1007/bfb0014052>
- [23] Gödel, K., *Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes*, *Dialectica* **12**, pages 280–287 (1958).
<https://doi.org/10.1111/j.1746-8361.1958.tb01464.x>
- [24] Gratzer, D., *Syntax and semantics of modal type theory*, Ph.D. thesis, Aarhus University (2023).
<https://pure.au.dk/portal/en/publications/syntax-and-semantics-of-modal-type-theory>
- [25] Gratzer, D., G. A. Kavvos, A. Nuyts and L. Birkedal, *Multimodal dependent type theory*, *Log. Methods Comput. Sci.* **17** (2021).
[https://doi.org/10.46298/LMCS-17\(3:11\)2021](https://doi.org/10.46298/LMCS-17(3:11)2021)
- [26] Hofmann, M. and T. Streicher, *The groupoid interpretation of type theory*, in: *Twenty-five years of constructive type theory (Venice, 1995)*, volume 36 of *Oxford Logic Guides*, pages 83–111, Oxford Univ. Press, New York (1998).
<https://doi.org/10.1093/oso/9780198501275.003.0008>
- [27] Jacobs, B., *Categorical logic and type theory*, volume 141 of *Studies in Logic and the Foundations of Mathematics*, North-Holland Publishing Co., Amsterdam (1999), ISBN 0-444-50170-3.
- [28] Kaposi, A., S. Huber and C. Sattler, *Gluing for type theory*, in: *4th International Conference on Formal Structures for Computation and Deduction (FSCD 2019)*, volume 131 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 25:1–25:19, Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2019).
<https://doi.org/10.4230/LIPICS.FSCD.2019.25>
- [29] Kapulkin, K. and P. L. Lumsdaine, *The simplicial model of univalent foundations (after Voevodsky)*, *J. Eur. Math. Soc. (JEMS)* **23**, pages 2071–2126 (2021), ISSN 1435-9855,1435-9863.
<https://doi.org/10.4171/JEMS/1050>
- [30] Kovács, A., *polynomial-model* (2020). Agda formalization of construction by Von Glehn [50].
<https://github.com/AndrasKovacs/polynomial-model>
- [31] Kovács, A., *antifunext* (2024). Agda formalization of construction by Pédrot and Tabareau [35].
<https://github.com/AndrasKovacs/antifunext>
- [32] Moss, S. K., *The Dialectica Models of Type Theory*, Ph.D. thesis, University of Cambridge (2018).
<https://doi.org/10.17863/CAM.28036>
- [33] Moss, S. K. and T. von Glehn, *Dialectica models of type theory*, in: A. Dawar and E. Grädel, editors, *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, pages 739–748, ACM (2018).
<https://doi.org/10.1145/3209108.3209207>
- [34] Orton, I. and A. M. Pitts, *Axioms for modelling cubical type theory in a topos*, *Logical Methods in Computer Science* **14** (2018).
[https://doi.org/10.23638/lmcs-14\(4:23\)2018](https://doi.org/10.23638/lmcs-14(4:23)2018)

- [35] Pédrot, P. and N. Tabareau, *Failure is not an option: An exceptional type theory*, in: A. Ahmed, editor, *Programming Languages and Systems - 27th European Symposium on Programming (ESOP 2018), Held as Part of the European Joint Conferences on Theory and Practice of Software (ETAPS 2018)*, volume 10801 of *Lecture Notes in Computer Science*, pages 245–271, Springer (2018), ISBN 978-3-319-89883-4.
https://doi.org/10.1007/978-3-319-89884-1_9
- [36] Rijke, E., *Introduction to Homotopy Type Theory*, Cambridge Studies in Advanced Mathematics, Cambridge University Press (2025), ISBN 9781108933568.
<https://doi.org/10.1017/9781108933568>
- [37] Scherer, G., *Deciding equivalence with sums and the empty type*, in: *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL)*, pages 374–386, Association for Computing Machinery (ACM) (2017).
<https://doi.org/10.1145/3009837.3009901>
- [38] Shulman, M., *Universal properties without function extensionality* (2014). Blog post on the Homotopy Type Theory blog.
<https://homotopytypetheory.org/2014/11/02/universal-properties-without-function-extensionality/>
- [39] Shulman, M., *All $(\infty, 1)$ -toposes have strict univalent universes* (2019). Preprint.
<https://arxiv.org/abs/1904.07004>
- [40] Shulman, M., A. Kovács et al., *Strong eta-rules for functions on sum types*, Proof Assistants StackExchange (2022). Version: 2022-12-10.
<https://proofassistants.stackexchange.com/q/1885>
- [41] Speight, S. L., *Groupoidal realizability for intensional type theory*, *Mathematical Structures in Computer Science* **34**, pages 911–944 (2024).
<https://doi.org/10.1017/s0960129524000343>
- [42] Streicher, T., *Investigations into Intensional Type Theory*, Habilitation thesis, Ludwig-Maximilians-Universität München (1993).
<https://www2.mathematik.tu-darmstadt.de/~streicher/HabilStreicher.pdf>
- [43] Swan, A. W., *A categorical formulation of Kraus’ paradox* (2024). Preprint.
<https://arxiv.org/abs/2403.17961>
- [44] The Univalent Foundations Program, *Homotopy Type Theory: Univalent Foundations of Mathematics*, Institute for Advanced Study (2013).
<https://homotopytypetheory.org/book>
- [45] Uemura, T., *Cubical assemblies, a univalent and impredicative universe and a failure of propositional resizing*, in: P. Dybjer, J. E. Santo and L. Pinto, editors, *24th International Conference on Types for Proofs and Programs (TYPES 2018)*, volume 130 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 7:1–7:20, Schloss Dagstuhl–Leibniz-Zentrum für Informatik (2019).
<https://doi.org/10.4230/LIPIcs.TYPES.2018.7>
- [46] van den Berg, B., *Univalent polymorphism*, *Annals of Pure and Applied Logic* **171**, page 102793 (2020).
<https://doi.org/10.1016/j.apal.2020.102793>
- [47] Voevodsky, V., *Univalence axiom and functional extensionality*.
https://github.com/UniMath/Foundations/blob/master/Proof_of_Extensionality/funextfun.v
- [48] Voevodsky, V., *Univalent Foundations Project (a modified version of an NSF grant application)* (2010).
https://www.math.ias.edu/vladimir/sites/math.ias.edu.vladimir/files/univalent_foundations_project.pdf
- [49] Voevodsky, V. et al., *coinductives* (2014). Mailing list discussion.
<https://groups.google.com/g/homotopytypetheory/c/tYRTcI20pyo/m/PIrI6t5me-oJ>
- [50] Von Glehn, T., *Polynomials and models of type theory*, Ph.D. thesis, University of Cambridge (2015).
<https://doi.org/10.17863/CAM.16245>
- [51] Winterhalter, T., *Formalisation and Meta-Theory of Type Theory*, Ph.D. thesis, Université de Nantes (2020).
<https://theses.hal.science/tel-05425836>