AN INDUCTIVE-RECURSIVE UNIVERSE GENERIC FOR SMALL FAMILIES

DANIEL GRATZER

Abstract. We show that it is possible to construct a universe in all Grothendieck topoi with injective codes à la Pujet and Tabareau [PT22] which is nonetheless generic for small families. As a trivial consequence, we show that $\mathtt{TT}_{\mathtt{obs}}$ admits interpretations in Grothendieck topoi suitable for use as internal languages.

Contents

Remark 1. We shall assume the Grothendieck universe axiom throughout this note to ensure a plentiful supply of Grothendieck universes.

In recent work, Pujet and Tabareau [PT22] have provided a comprehensive *observational type theory* complete with a hierarchy of universes and proven their theory enjoys decidable type-checking and a number of other pleasant results. In order to ensure this, *op. cit.* requires a number of counter-intuitive properties of the universes. Specifically, they require that the type-constructors on the universe are injective. In the case of dependent products, this means that given a proof $e : \prod_{a:A_0} B_0(a) \sim \prod_{a:A_1} B_1(a)$, one can *always* produce a pair of proofs:

$$e_0 : A_0 \sim A_1 \qquad\qquad e_1 : \textstyle\prod_{a:A_0} B_0(a) \sim B_1(\mathsf{cast}(A_0, A_1, e_0, a))$$

In other words, they require that $\prod_{-} -$ is injective.

Semantically, this is far from natural. Imagine, for instance, that $A_0 = A_1 = \mathbf{0}$, so that both function types are equivalent to $\mathbf{1}$, regardless of the choice of B_0 or B_1. One can easily construct a model where these types are identified, so that we have no hope of producing e_1 in such a model. In fact, this small example already shows that $\mathtt{TT}_{\mathtt{obs}}$ cannot be given the standard set-theoretic model wherein the universe is realized by a Grothendieck universe. Pujet and Tabareau [PT22], however, have shown that $\mathtt{TT}_{\mathtt{obs}}$ admits a model in setoids by using an inductive-recursive construction to model the universe. We show that this approach is easily generalized to give a model of $\mathtt{TT}_{\mathtt{obs}}$ in arbitrary Grothendieck topoi and that a

Date: Friday 11^{th} February, 2022.

simple modification to the standard IR universe ensures that $\mathtt{TT}_{\mathtt{obs}}$ forms the basis for a workable internal language in all of these settings.

Remark 2. In a concession to brevity and convenience we modify $\mathtt{TT}_{\mathtt{obs}}$ in several ways to better fit our techniques. Firstly, we regard the type theory as a signature in some logical framework (generalized algebraic theories, QIITs, representable map categories, LCCCs, or the like) thereby drop all discussions of coherence and partial interpretation functions featured prominently in Pujet and Tabareau [PT22].

Secondly, we work with the universes as *strict à la Tarski universes*. Without this change its inconceivable to have models of the theory in more complex categories where the distinction between objects and morphisms cannot be blurred away. For a user, however, the gap is substantially smaller than one might fear. We can always add a largest U_ω universe à la Tarski and systematically replace genuine types in a program with codes in this universe. Even this change is unnecessary however, as a type-directed elaboration procedure can easily paper over the mismatches.

Remark 3. We have occasion in this note to discuss both strong and weak Tarski universes. A strong Tarski universe is the standard notion: a type U and an explicitly decoding function El which commutes with a choice of codes in U for dependent products, sums, etc. A weak Tarski universe requires the same operations, but only satisfies the commutativity conditions up to isomorphism. The latter tends to more natural to obtain categorically, and some implementation-work has shown the notion to be workable in practice [Red20].

1. Modeling $\mathtt{TT}_{\mathtt{obs}}$ through induction-recursion

One can model $\mathtt{TT}_{\mathtt{obs}}$ in **Set** by interpreting the universe not by a Grothendieck universe, but instead by an *inductive-recursive* (IR) universe [Dyb00]. For our purposes, we will focus on *small* induction [Han+13], where the eliminator is valued in a universe of types smaller than the inductive definition. More verbosely, a small inductive-recursive definition is a pair of some inductively defined family $A : \mathsf{U}_1$ defined simultaneously with a function $r : A \longrightarrow \mathsf{U}_0$. Importantly, while induction-recursion generally has remarkable proof-theoretic strength, small induction-recursion is a fairly innocuous reasoning principle and can be encoded in extensional type theory with indexed inductive types.

Importantly, small induction-recursion is still sufficient to define IR universes in a type theory with universes:

$$
\begin{array}{l}
\mathbf{data}\ V : \mathsf{U}_1\ \mathbf{where} \\
\quad \mathtt{bool} : V \\
\quad \mathtt{unit} : V \\
\quad \mathtt{pi} : (A : V) \to (r(A) \to V) \to V \\
\quad \mathtt{sg} : (A : V) \to (r(A) \to V) \to V
\end{array}
$$

$$
\begin{array}{ll}
r : V \to \mathsf{U}_0 & \\
r(\mathtt{bool}) = \mathbf{2} & r(\mathtt{unit}) = \mathbf{1} \\
r(\mathtt{pi}(A, B)) = \prod_{a:r(A)} r(B(a)) & r(\mathtt{sg}(A, B)) = \sum_{a:r(A)} r(B(a))
\end{array}
$$

As an inductive type V admits an induction principle which we can use to prove that $\mathtt{pi}$ is injective, just as we can show that the successor is injective. Consequently, V provides exactly the basis we need to interpret $\mathtt{TT}_{\mathtt{obs}}$ into **Set**.

Summarizing, to interpret $\mathtt{TT}_{\mathtt{obs}}$ into **Set** we start with some Grothendieck universe $\mathcal{U}$ and we then use small IR within $\mathcal{U}$ to define a new set V equipped with injective codes for the type-constructors, and use this new set V to interpret the universe of $\mathtt{TT}_{\mathtt{obs}}$. In fact, because small induction-recursion lifts to Grothendieck topoi, this same approach yields an interpretation of $\mathtt{TT}_{\mathtt{obs}}$ into arbitrary topoi.

1.1. **The problem with V.** While this process yields a workable model of $\mathtt{TT}_{\mathtt{obs}}$, the model does not form the basis of a good *internal language*. In particular, because V is defined by explicitly enumerating the various constructors of the universe, V lacks codes representing objects of the model laying outside the image of the interpretation function. To pick a specific example, consider attempting replaying the construction of Orton and Pitts [OP18] in $\mathtt{TT}_{\mathtt{obs}}$. We could not specialize the model above to **cSet** to justify this development, because they require the universe to contain an interval object and V simply does not include such a constructor.

Of course, we could specialize the model in cubical sets further and explicitly include an interval code to the definition of V. This is, however, hardly a satisfactory state of affairs! We do not want a foundation for using $\mathtt{TT}_{\mathtt{obs}}$ as an internal language that needs to be changed every time we use a new aspect of our model.

We can quantify the problem more precisely by shifting our perspective on universes. While in type theory a universe is a particular pair of a type and a family dependent over that type, in category theory a universe is a collection of maps $\mathcal{S}$ stable under pullback and closed under various operations [Str05]. One also requires a *generic family* for such a class—this is the categorical equivalent of what type theorists call a universe—but generic families are not defined up to isomorphism and are not an invariant characteristic of universes.

We can phrase our issue with the IR universe (V, r) somewhat more precisely by saying that it is generic for a class $\mathcal{T}$ which lacks many important families in **cSet**. In fact, a family is classified by (V, r) only if it lies in the essential image of the unique functor logical $\mathcal{I} \longrightarrow \mathbf{cSet}$, where $\mathcal{I}$ is the initial elementary topos with a natural number object. This is clearly an issue if we aim to use the universe to axiomatize types specific to **cSet** or indeed any topos $\mathcal{E}$.

1.2. **A plausible solution.** Of course, no matter how we interpret the universe some families in $\mathcal{E}$ will lay outside it. Indeed, for set-theoretical reasons we cannot hope to find a universe containing all families in $\mathcal{E}$, but we can hope for the next best alternative: a universe which contains all 'small' families.

The gold standard in this regard for Grothendieck topoi is to have a universe of all *relatively κ-compact* families [Shu19], where κ is some inaccessible cardinal. In fact, given a hierarchy of such universes for ever-increasing κ, we can ensure that every family lies within some universe.[1] Helpfully, Streicher [Str05] shows that for all sufficiently large κ this universe satisfies all the desirable axioms. Crucially, *op. cit.* shows that a generic family for the class of relatively κ-compact morphisms exists in all Grothendieck topoi. Unfortunately, the supplied generic family is based upon Grothendieck universes in **Set**—precisely the generic family we just argued cannot be used to interpret $\mathtt{TT}_{\mathtt{obs}}$.

Fortunately, generic families are not uniquely determined by a universe, and so we can hope for a better one for the same class of morphisms:

[1] The Grothendieck universe axiom essentially stipulates this to be the case for **Set**.

Conjecture 1.1. *There is a generic families for relatively κ-compact morphisms equipped with injective codes for dependent products, sums, etc. in an arbitrary Grothendieck topos.*

2. A generic family defined by small induction-recursion

Fix some Grothendieck topos $\mathcal{E}$ and a pair of inaccessible cardinals $\kappa_0 < \kappa_1$. Streicher [Str05] ensures that relatively κ_i-compact families organize into universes $\mathcal{S}_{\kappa_i}$ in $\mathcal{E}$ with generic families $\tau_i : \mathcal{U}_i^\bullet \longrightarrow \mathcal{U}_i$ and inspection on the construction of the generic families reveals that $\mathcal{U}_0 \longrightarrow \mathbf{1}$ is relatively κ_1-compact family. We will now construct a new generic family for $\mathcal{S}_{\kappa_0}$ along with injective codes closing it under dependent products, sums, etc.

To make this process a bit more fluid, we work in the *internal language* of $\mathcal{E}$. That is, we work with an extensional type theory with a hierarchy of two weak universes à la Tarski $\mathsf{U}_0 : \mathsf{U}_1$.[2] We will construct a universe $(V : \mathsf{U}_1, \mathsf{El}_V(-) : V \to \mathsf{U}_0)$ with the following operations:

- $\mathtt{up} : \mathsf{U}_0 \longrightarrow V$ such that $\mathsf{id} = \mathsf{El}_V \circ \mathtt{up}$.
- $\mathtt{pi} : \prod_{a:V} \prod_{b:\mathsf{El}_V(a) \to V} V$ such that $\mathtt{pi}$ is injective and $\mathsf{El}(\mathtt{pi}(a,b)) = \prod_{x:\mathsf{El}(a)} \mathsf{El}(b(x))$.

Remark 4. In fact, V can trivially be extended to enjoy injective constructors similar to $\mathtt{pi}$ for dependent sums, booleans, equality types, etc. but we will focus on dependent products as a representative example.

The first of these requirements ensures that (V, El_V) is generic for at least as many maps as U_0 and the second gives the desired injective code for close V under dependent products. In fact, since U_0 is generic for a class of maps *already* closed under dependent products (though it does not necessarily witness this fact by an injective code) we can conclude that V is generic for precisely the same class as U_0.

Let us define V by the following (small) inductive-recursive definition:

$$\begin{array}{l}\textbf{data } V : \mathsf{U}_1 \textbf{ where}\\ \quad \mathtt{up} : \mathsf{U}_0 \to V\\ \quad \mathtt{pi} : (A : V) \to (\mathsf{El}_V(A) \to V) \to V\end{array}$$

$$\mathsf{El}_V(\mathtt{up}(A)) = A \qquad \mathsf{El}_V(\mathtt{pi}(A,B)) = \prod_{a:\mathsf{El}_V(A)} \mathsf{El}_V(B(a))$$

The two required functions are now just constructors of V and both satisfy the required properties simply by definition of El_V. Already from this simple construction we conclude the following:

Theorem 2.1. *In an arbitrary Grothendieck topos $\mathcal{E}$, there exists an interpretation of $\mathtt{TT_{obs}}$ with one weak universe à la Tarksi which sends the universe to a generic family for relatively κ_0-compact families.*

Remark 5. Notice here that we have obtained only a *weak* universe, because $\mathtt{pi}(A, B)$ decodes to a dependent product in U_0 which then must be lifted to U_1 to be regarded as a type in our model. Unfortunately, we have not assumed that code witnessing closure under dependent products in U_0 lifts to the equivalent code in U_1, and so do not obtain a model satisfying this equation. If we had assumed this however—and this requirement is satisfied by e.g. the generic family supplied by Hofmann and Streicher [HS97]—we could correspondingly strengthen Theorem 2.1.

[2] We ignore strictness issues here, which can be rectified through any number of well-known constructions.

3. A strictly cumulative hierarchy

Theorem 2.1 is an excellent starting point, but we are interested in a hierarchy of such universes. As before, we will show that we can 'correct' a universe without injective codes to a one with injective codes. We work in an arbitrary Grothendieck topos $\mathcal{E}$. We work this time with a hierarchy of inaccessible cardinals $\kappa_0 < \kappa_1 < \dots$. These induce a hierarchy of universes $\mathsf{U}_0 : \mathsf{U}_1 : \dots$ in the extensional type theory of $\mathcal{E}$, but unlike Section 2, we will assume that we have constructed this hierarchy to be strictly cumulative. This can be done in presheaf topoi using the construction of Hofmann and Streicher [HS97]. In a general Grothendieck topos, one can use a more complex construction of Shulman [Shu15], which is discussed at length in forthcoming work by Gratzer, Shulman, and Sterling.

We now proceed to inductively replace U_i by V_i such that V_i is equipped with an injective operation $\mathtt{pi}_i : \prod_{A:V_i}(\mathsf{El}_{V_i}(A) \to V_i) \to V_i$ and (V_i, El_{V_i}) is generic for the same class of types as U_i, just as in Section 2. We further ensure that there is an element $\mathtt{uni}_i : V_j$ for all $i < j$ such that $\mathsf{El}_{V_j}(\mathtt{uni}_i) = V_i$

Assume that $(V_k : \mathsf{U}_{k+1}, \mathsf{El}_{V_k} : V_k \to \mathsf{U}_k)$ has been defined for all $k < i$. We define V_i and El_{V_i} as follows using small induction-recursion in U_{k+1}:

$$
\begin{array}{l}
\textbf{data } V_i : \mathsf{U}_{i+1} \textbf{ where} \\
\quad \mathtt{up} : \mathsf{U}_i \to V_i \\
\quad \mathtt{uni}_0, \cdots, \mathtt{uni}_{i-1} : V_i \\
\quad \mathtt{pi} : (A : V_i) \to (\mathsf{El}_{V_i}(A) \to V_i) \to V_i
\end{array}
$$

$$
\mathsf{El}_{V_i}(\mathtt{up}(A)) = A \qquad \mathsf{El}_{V_i}(\mathtt{uni}_k) = V_k \qquad \mathsf{El}_{V_i}(\mathtt{pi}(A,B)) = \textstyle\prod_{a:\mathsf{El}_{V_i}(A)} \mathsf{El}_{V_i}(B(a))
$$

It is plain that V_i satisfies the required properties. As a final step, for each $i < j$ we define a function $\mathsf{lift} : V_i \longrightarrow V_j$ such that $\mathsf{El}_{V_i}(A) = \mathsf{El}_{V_j}(\mathsf{lift}(A))$ and so that lift commutes with $\mathtt{pi}$ and $\mathtt{uni}_k$. In fact, this specification fully defines lift and directly translates into a definition using the induction principle for V_i:

$$
\begin{aligned}
&\mathsf{lift}(\mathtt{up}(A)) = \mathtt{up}(\uparrow A) \\
&\mathsf{lift}(\mathtt{pi}(A,B)) = \mathtt{pi}(\mathsf{lift}(A), \mathsf{lift} \circ B) \\
&\mathsf{lift}(\mathtt{uni}_k) = \mathtt{uni}_k
\end{aligned}
$$

Inspection shows that lift is functorial, and we thereby obtain the required strictly cumulative hierarchy of universes.

Theorem 3.1. *In an arbitrary Grothendieck topos $\mathcal{E}$, there exists an interpretation of $\mathtt{TT}_{\mathtt{obs}}$ with cumulative countable hierarchy of universes such that the ith universe is sent to a generic family for relatively κ_i-compact families.*

In fact, we have really proven the following more general result:

Theorem 3.2. *A model of type theory with a cumulative hierarchy also supports a hierarchy with injective codes which remains generic for the same universes.*

4. Cumulativity from weak universes and induction-recursion

Thus far our constructions have used only *small* induction-recursion, so that the decoding function associated with the inductive type targets a lower universe level. This restriction ensures that the process can be decoded to indexed inductive types. If we assume, however, that we are working in a model which supports true induction-recursion we can replicate Theorem 3.1 without assuming the input universe hierarchy is strictly cumulative.

We feel this construction is potentially interesting for constructivists; a constructively acceptable version of the universes introduced by Shulman [Shu15] remains elusive, and so a strictly cumulative hierarchy of universes in arbitrary Grothendieck topoi presently requires choice. A priori, the same might not be true for induction-recursion and Streicher [Str05] has already shown that a hierarchy of universes which is merely *weakly* cumulative exists constructively. Accordingly, this construction offers an interesting line of attack for a constructively acceptable hierarchy of universes in all Grothendieck topoi.

Let us fix a hierarchy of weak Tarski universes $\mathsf{U}_0 : \cdots : \mathsf{U}_\omega$. We proceed as before and inductively replace U_i by V_i so that the latter equips the former with a strict choice of codes. Unlike in Section 3, we do not use small induction-recursion in U_{i+1} in order to carry out this construction. Instead we use large IR in U_ω each time, and thereby avoid the need for a coherent choice of connectives in U_i.[3]

$$\begin{array}{l}\textbf{data } V_i : \mathsf{U}_\omega \textbf{ where}\\ \quad \mathtt{up} : \mathsf{U}_i \to V_i\\ \quad \mathtt{uni}_0, \cdots, \mathtt{uni}_{i-1} : V_i\\ \quad \mathtt{pi} : (A : V_i) \to (\mathsf{El}_{V_i}(A) \to V_i) \to V_i\end{array}$$

$$\mathsf{El}_{V_i}(\mathtt{up}(A)) = A \qquad \mathsf{El}_{V_i}(\mathtt{uni}_k) = V_k \qquad \mathsf{El}_{V_i}(\mathtt{pi}(A, B)) = \prod_{a:\mathsf{El}_{V_i}(A)} \mathsf{El}_{V_i}(B(a))$$

The lifting operation is define *mutatis mutandis*.

Theorem 4.1. *A model with a weak hierarchy and induction-recursion can be extended to support a strict hierarchy generic for the same universes.*

We emphasize the last point of this statement. It is well-known that large induction-recursion is sufficient to define a cumulative hierarchy—this was the original example of IR—but we have shown that our `up` trick is sufficient to define a cumulative hierarchy which remains generic for e.g., relatively κ-compact families. This point is unremarkable from within the type theory itself, but crucial when using type theory as an internal language; it ensures that our universes actually contain interesting families specific to a model.

Acknowledgments

I am grateful for conversations with Carlo Angiuli and Jonathan Sterling.

References

[Dyb00] Peter Dybjer. "A general formulation of simultaneous inductive-recursive definitions in type theory". In: *Journal of Symbolic Logic* 65.2 (2000), pp. 525–549. DOI: 10.2307/2586554 (cit. on p. 2).

[Han+13] Peter Hancock, Conor McBride, Neil Ghani, Lorenzo Malatesta, and Thorsten Altenkirch. "Small Induction Recursion". In: *Typed Lambda Calculi and Applications*. Ed. by Masahito Hasegawa. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 156–172. ISBN: 978-3-642-38946-7 (cit. on p. 2).

[3]In fact, we do not even require that U_i be closed under *any* connectives in this construction. We are freely closing the universe classified by U_i with dependent products; if U_i was already closed under dependent products this is an idempotent operation.

[HS97] Martin Hofmann and Thomas Streicher. "Lifting Grothendieck Universes". Unpublished note. 1997. URL: https://www2.mathematik.tu-darmstadt.de/~streicher/NOTES/lift.pdf (cit. on pp. 4, 5).

[OP18] Ian Orton and Andrew M. Pitts. "Axioms for Modelling Cubical Type Theory in a Topos". In: *Logical Methods in Computer Science* 14.4 (2018). DOI: 10.23638/LMCS-14(4:23)2018. arXiv: 1712.04864 (cit. on p. 3).

[PT22] Loïc Pujet and Nicolas Tabareau. "Observational Equality: Now for Good". In: *Proc. ACM Program. Lang.* 6.POPL (Dec. 2022). DOI: 10.1145/3498693. URL: https://doi.org/10.1145/3498693 (cit. on pp. 1, 2).

[Red20] The RedPRL Development Team. *cooltt*. 2020. URL: http://www.github.com/RedPRL/cooltt (cit. on p. 2).

[Shu15] Michael Shulman. "The Univalence Axiom for Elegant Reedy Presheaves". In: *Homology, Homotopy and Applications* 17 (2 2015), pp. 81–106 (cit. on pp. 5, 6).

[Shu19] Michael Shulman. *All $(\infty, 1)$-toposes have strict univalent universes*. 2019. arXiv: 1904.07004 [math.AT] (cit. on p. 3).

[Str05] Thomas Streicher. "Universes in toposes". In: *From Sets and Types to Topology and Analysis: Towards practical foundations for constructive mathematics*. Ed. by Laura Crosilla and Peter Schuster. Vol. 48. Oxford Logical Guides. Oxford: Oxford University Press, 2005, pp. 78–90. ISBN: 978-0-19-856651-9. DOI: 10.1093/acprof:oso/9780198566519.001.0001 (cit. on pp. 3, 4, 6).