Universe oddities

Martín Hötzel Escardó

Work in progress with

Marc Bezem, Thierry Coquand and Peter Dybjer

University of Birmingham, UK

Foundations and Applications of Univalent Mathematics
18-20 December 2019, Herrsching, Germany, 2019

I'll begin with an example

Define the type of magmas to be Magma $:= \sum_{A:\mathcal{U}}$ is set $A \times (A \times A \to A)$.

Take the primitive notion of equivalence to be *map with contractible fibers*.

Theorem. Univalence implies that magma identity is in canonical equivalence with magma isomorphism.

Spot the mistake in the following proof

Proof sketch.

1. Because identity functions are trivially half-adjoint equivalences, all equivalences are half adjoint equivalences, by univalence.

2. (Change of variable.) If $f : X \to Y$ is a half adjoint equivalence, and $A : Y \to \mathcal{U}$ is a type family, then $\Sigma(y : Y), A\, y \simeq \Sigma(x : X), A\,(f\, x)$.

3. For magmas $\mathcal{A} = (A, i_A, \cdot_A)$ and $\mathcal{B} = (B, i_B, \cdot_B)$

$$\begin{aligned} \mathcal{A} = \mathcal{B} &\simeq \Sigma(p : A = B), \text{transport } p\,(- \cdot_A -) = (- \cdot_B -) \\ &\simeq \Sigma(p : A = B), \text{the equivalence } A \to B \text{ induced by } p \text{ is a homomorphism} \\ &\simeq \Sigma(h : A \simeq B), h \text{ is a homomorphism,} \end{aligned}$$

where the last equivalence is by change of variable along the canonical equivalence $A = B \to A \simeq B$.

Each step is individually correct

1. Because identity functions are half-adjoint equivalences, all equivalences are trivially half adjoint equivalences, by univalence.

2. (Change of variable.) If $f : X \to Y$ is a half adjoint equivalence, and $A : Y \to \mathcal{U}$ is a type family, then $\Sigma(y : Y), A\,y \simeq \Sigma(x : X), A\,(f\,x)$.

3. For magmas $\mathcal{A} = (A, i_A, \cdot_A)$ and $\mathcal{B} = (B, i_B, \cdot_B)$

$$\begin{aligned} \mathcal{A} = \mathcal{B} &\simeq \Sigma(p : A = B), \text{transport } p\,(- \cdot_A -) = (- \cdot_B -) \\ &\simeq \Sigma(p : A = B), \text{the equivalence } A \to B \text{ induced by } p \text{ is a homomorphism} \\ &\simeq \Sigma(h : A \simeq B), h \text{ is a homomorphism,} \end{aligned}$$

where the last equivalence is by change of variable along the canonical equivalence $A = B \to A \simeq B$.

Agda and Coq reject this proof

- And so does Book HoTT.
- But it would be accepted with type-in-type.
- (And hence UniMath would accept it.)

The problem with the proof

1. Typical ambiguity.
2. We pretend there is only one universe in our discourse.
3. The universe levels are left implicit.
4. The theorem that

   if all identitities satisfy a condition then so do all equivalences

   applies to equivalences between types of the *same* universe.
5. But the canonical equivalence $A = B \to A \simeq B$ relates types of *different* universes.
6. Coq gives the error message "universe inconsistency" when we try to prove that this canonical equivalence is a half-adjoint equivalence by equivalence induction. (Kindly checked by Mike Shulman for me.)

There is a problem with $(A = B) = (A \simeq B)$ too

1. By definition of univalence, we do get $(A = B) \simeq (A \simeq B)$.
2. However, the types $A = B$ and $A \simeq B$ live in different universes.
3. Without universe cumulativity, $(A = B) = (A \simeq B)$ doesn't even type check.
4. With universe cumulativity, this can be written down and is true.
5. But *not* by a direct application of the definition of univalence.
6. This requires proof.
7. Moreover, it requires univalence not only of the universe where the types A and B live, but also of the next one.
8. This is because a priori the identity type $A = B$ depends not only on A and B but also on their universe.
9. The types $A =_{\mathcal{U}_n} B$ and $A =_{\mathcal{U}_{n+1}} B$ are not the same on the nose, even if they both live in $\mathcal{U}_{n+2}$.
10. With univalence, we can prove that the inclusion of any universe into a larger universe is an embedding, which makes the equation true.

Agda's universe polymorphism

1. Agda's universes are called Set_i.
2. This terminology is not very appropriate for univalent mathematics, as their elements need not be sets,
3. I will rename them $\mathcal{U}_i$ for the purposes of this talk.
4. There is a type Level, in the first universe, where the universe indices live.
5. We have a level zero, and successor operation, a binary maximum operation, but no eliminator.
6. Each type lives in a unique universe.
7. We can define the empty type of the first (or any specific) universe.
8. Or we can define version parametrized by a level.
9. If $X : \mathcal{U}_i$ and $Y : U_j$ then $X \times Y$ and $X + Y$ live in $\mathcal{U}_{\max(i,j)}$.
10. If $X : \mathcal{U}_j$ and $A : X \to \mathcal{U}_j$ then $\Pi\, X\, A$ and $\Sigma\, X\, A$ live in $\mathcal{U}_{\max(i,j)}$.
11. There is a "sort" $\mathcal{U}_\omega$, but ω is not an element of the type Level.

Examples

1. is univalent : $(i : \mathsf{Level}) \to \mathcal{U}_{i+1}$

2. Univalence : $\mathcal{U}_\omega$
   Univalence $= (i : \mathsf{Level}) \to$ is univalent i

- So this goes beyond the type theory of the HoTT Book, in which universe levels are meta-linguistic, and in which univalence is an axiom scheme.

Agda's universe polymorphism continued

1. There is an inclusion function $\mathsf{Lift} : \{i\, j : \mathsf{Level}\} \to \mathcal{U}_i \to \mathcal{U}_{\max(i,j)}$.

   (Almost never needed or used.)

2. Together with functions $\mathsf{lift} : X \to \mathsf{Lift}\, X$ and $\mathsf{lower} : \mathsf{Lift}\, X \to X$ which are definitionally mutually inverse.

3. With univalence, we can prove that Lift is an embedding of $\mathcal{U}_i$ into $\mathcal{U}_{\max(i,j)}$.

4. The univalence of the two universes is needed.

5. This means that there is a (canonical) equivalence between $A =_{\mathcal{U}_i} B$ and $\mathsf{Lift}\, A =_{\mathcal{U}_{\max(i,j)}} \mathsf{Lift}\, B$.

   The map Lift preserves and reflects identity.

6. Hence $\mathsf{Lift}(A =_{\mathcal{U}_i} B) =_{\mathcal{U}_{\max(i,j)+1}} (\mathsf{Lift}\, A =_{\mathcal{U}_{\max(i,j)}} \mathsf{Lift}\, B)$.

Upwards and downwads equivalence induction

Fix three universe levels i, j, k.

1. (Upwards induction.) Assume $\mathcal{U}_{\max(i,j)}$ is univalent.
   - Let $A : (X : \mathcal{U}_i)\,(Y : \mathcal{U}_{\max(i,j)}) \to X \simeq Y \to U_k$.
   - Denote by L_X the equivalence $X \simeq \mathsf{Lift}\, X$ induced by the maps $\mathsf{lift}, \mathsf{lower}$.
   - If $A\,X\,(\mathsf{Lift}\,X)\,L_X$ for all $X : \mathcal{U}_i$, then $A\,X\;Y\;e$ for all $X : \mathcal{U}_i$, $Y : \mathcal{U}_{\max(i,j)}$ and $e : X \simeq Y$.
   - Moreover, the evident "computation rule" holds up to identity.

2. (Downwards induction.) Dual, with X ranging over $\mathcal{U}_{\max(i,j)}$ and Y over $\mathcal{U}_i$.

Odd situation

1. We can prove things about all equivalences going up universes by induction.
2. We can prove things about all equivalences going down universes by induction.
3. Every equivalence goes up or down.
4. But we cannot prove things for arbitrary equivalences between arbitrary universes by induction.

Global properties

Examples of global properties include:

1. X is a proposition.

2. X is a set.

In Agda's type system:

$$\begin{array}{lcl} \text{global-property} & : & \mathcal{U}_\omega \\ \text{global-property} & = & \{i : \text{Level}\} \to \mathcal{U}_i \to \mathcal{U}_i \end{array}$$

Of course one can think of many useful variations.

Are global properties invariant under equivalence?

1. Counterexample: $A\{i\}X = \mathrm{Lift}(i = 0)$.
2. We could try to blame Agda's ability to compare levels for equality.
3. But Mike Shulman first suggested this counter-example to us in *models*.

Say that a global property A is *cumulative* if we have an equivalence $A\,X \simeq A\,(\mathrm{Lift}\,X)$ for any type X.

Fact. A global property is invariant under equivalence if and only if it is cumulative.

One more oddity

Consider the following type:

$$(A \times B) = (C \times A) \to (B \times A) = (C \times A)$$

There are three incomparable maximally general universe level assignments for the types A, B, C in a non-cumulative system

But in a cumulativity system, there is a unique most general assignment.

Coq's universe polymorphism

I am not entirely sure how it works. The main differences are

1. Cumulativity.
2. Typical ambiguity.
3. Use of universe constraints and a constraint solver.
4. Absence of a type of levels.

But I believe it should be possible to reproduce the above facts proved in Agda (I invite you to try if you know enough Coq).

Some proposed rules for universe polymorphism

We have a draft paper.

1. With external or internal levels, and level judgments rather than a type of levels.
2. With or without cumulativity.
3. New algorithm for generating and solving constraints (typical ambiguity with cumulativity).

Related work: Harper, Pollack, Chan, Huet, Assaf, Voevodsky.

We noticed the above oddities when developing

1. https://www.cs.bham.ac.uk/~mhe/agda-new/
2. https://www.cs.bham.ac.uk/~mhe/HoTT-UF-in-Agda-Lecture-Notes/