Expressing 'The Structure of' in Homotopy Type Theory

David Corfield, Department of Philosophy, University of Kent

15 January 2016

1 Introduction

Philosophers of mathematics have long discussed what is meant by the expression 'the structure of A' for a given mathematical entity A. For the structuralist, it is believed that, in some sense, mathematical entities simply *are* structures. For example, famously it is possible to give different constructions within set theory of sets which may be taken to represent the natural numbers. The structure common to these constructions is then understood to be what the natural numbers are. It is also thought that to locate the structure of any construction there needs to be a way to abstract it from whatever it is that 'carries' it, and conditions should be given for when two such abstracted structures are the same.

In very recent years homotopy type theory (HoTT) has emerged as a new foundational language which, it is claimed (Awodey 2014), via its so-called 'Univalence axiom', captures what it essentially right about the structuralist position. By use of an 'abstraction principle', Awodey informally defines a notion of structure through the isomorphism, or better equivalence, of types:

$$str(A) = str(B) \Leftrightarrow A \cong B. \qquad \text{(DS)}$$

He concludes, after a discussion of the Univalence axiom (UA), which says of types that if they are equivalent as defined in HoTT, then they may be considered equal:

> ...observe that, as an informal consequence of (UA), together with the very definition of "structure" (DS), we have that two mathematical objects are identical if and only if they have the same structure:
>
> $$str(A) = str(B) \Leftrightarrow A = B.$$
>
> In other words, mathematical objects simply *are* structures. Could there be a stronger formulation of structuralism? (Awodey 2014, p. 12)

In other words, taking HoTT as our foundation, all constructions are already fully structural.

This conclusion seems to me to be correct, but in this article, I shall adopt a different argument strategy by examining whether HoTT itself can tell us a little more about such locutions as 'the structure of A', 'A and B share the same structure' and 'places in the structure'. Rather than invoking the Fregean notion of an abstraction principle, I shall propose what appear to be the only plausible definitions within HoTT itself of the relevant terms.

To be in a position to do so, first we need to see how to understand the relevant employment of the word 'the'. As a cross-check we will see that my proposed 'introduction rule' for 'the A' makes sense of common forms of speech among mathematicians, such as '*the* product of A and B'. Next we will consider 'structure of' as applied to a type in the system. These ingredients are combined to treat the whole term 'the structure of A'. Finally I consider 'places in' such a structure, and extend these analyses to structured types. I shall assume a fair working knowledge of HoTT to keep this note to a reasonable length. Readers should refer to the book *Homotopy Type Theory* (Univalent Foundations Program, 2014).

2 'The'

We use 'the' in a number of somewhat related ways:

- The prime minister of the United Kingdom is right-handed.
- The Romans invaded Britain in 43 AD.
- The platypus is a noctural creature.

For the purposes of this note, I shall be considering its use in definite descriptions, as in the first of the examples above, where 'the' is followed by a singular noun, since this seems to be the case with 'the structure of A'.

From the perspective of HoTT, I take there to be two forms of definite description:

1. 'The A', where A is a type.
2. 'The $f(a)$', where $a : A$ and $f : B^A$, for some types A and B.

The former is properly formed when there is a unique individual of the type A. This may come about by forming a singleton subtype of an existing type, such as 'The highest mountain', that is, the unique element of the subtype of the type 'Mountain', which is 'Mountain for which no other mountain is higher'. Such subtypes may be formed within the type theory as what are called dependent sums. These are pairs formed of an element of the main type, here a mountain, along with a warrant that the specified condition holds, here a proof that this mountain is the highest.

In case (2), the fact that f is a function forces the existence and uniqueness of $f(a)$, such as in the expression 'the colour of my front door', where 'colour of' is a map from some type of objects to the type of colours. A simple extension would allow B to depend on A, so $f(a) : B(a)$. In either case, we might form, as in the paragraph above, a singleton subtype, such as 'Colour which is the colour of my front door', and in this way, we can reduce to (1), 'The A' for some type A.

Where I have been considering singleton types as though they are sets with one element, in HoTT types need not just be sets, but may be 'mere propositions' or may be 'higher groupoids'. Let us consider the latter first in the form of a type which appears to possess a variety of elements and yet where we are still inclined to apply 'the' to the type. It is well known that mathematicians will say 'the product of two sets' knowing full well that there are seemingly different ways to form such a product out of the elements of the sets. Category theory has explained how to think of this case as one where, when a construction has been defined by a *universal property*, it does not matter which representative one takes as product. This is because there is a *canonical* isomorphism between any two representatives. Category theory and type theory work hand-in-hand here. The universal nature of the product construction applied to all homotopy types as described by the former is perfectly captured in the combination of the four rules of type formation, term introduction, term elimination and computation.

We could say that the type of products of two sets is a groupoid in which objects are related by unique morphisms, or in other words that the groupoid is equivalent to the trivial groupoid. Up to equivalence, as a homotopy type, such a 'coarse' groupoid is equivalent to a singleton set. Taking our cue from topology, we name this property 'contractibility'.

Naturally, there is a construction in HoTT which defines what it is for a type to possess this property:

$$X : Type \vdash isContr(X) \equiv \sum_{x:\,X} \prod_{y:\,X} (y = x) : Prop.$$

This dependent sum requires us to produce an element of the type, and then a coherent collection of identities between that element and each element of the type. In the case of the type $Product(A, B)$ for two sets A and B, we have a representative $A \times B$, and for any other representative, the canonical isomorphism.

One level down, when X is a set (in the HoTT sense that the identity of two of its elements is a 'mere proposition'), then as expected we find that X is contractible precisely if it is a singleton. On the other hand, when X itself is a 'mere proposition', that is, a type where every two elements (if there are any) are equal, then contractibility amounts to X being inhabited, and so true.

Now the claim is that we should only form the term 'the X' for a given type X once we have established that $isContr(X)$ is inhabited. In case a type is empty, we should not say 'the X'. Russell's 'the present King of France' should never have been formed, let alone declared to be bald. Then we can describe the rule of what we might call *the introduction*:

$$X : Type, (x, p) : isContr(X) \vdash the(X, x, p) \equiv x : X,$$

where we typically do not think about the dependency on p. One explanation for ignoring p from the point of view of HoTT is that there can only be one such element since $isContr(X)$ is a mere proposition (Lemma 3.11.4 of the HoTT book).

So now we see that contractibility makes sense of our application of 'the' to an apparent groupoid such as 'the product of two types A and B'. One might think that there could be many ways to produce such a product, but the universal property defining what it means to be a product ensures that any candidate is uniquely isomorphic to any other.

It is perhaps illuminating then to consider on this reading that, strictly speaking, for a type which is a groupoid in which every pair of elements is isomorphic, but not canonically so, we should not apply 'the' to the type. We see this in the mathematical construction of algebraically closing a field, where there is a reluctance to say '*the* algebraic closure of field F' although all such closures are isomorphic, since they are not canonically so (Henriques 2010).

What of contractible, and so true, propositions? Well here while we do not prefix ‘the’ to a proposition such as ‘it is raining’, we certainly do say ‘the fact that it is raining’. It might be a good idea then to call any type which is a proposition ‘Fact that such and such’. So we would have ‘Fact that it is raining’ as a type, and, if it is inhabited, we would designate its element by the term ‘the fact that it is raining’.

Before moving on, let us note that under the interpretation of this section it becomes difficult to see how we could ever make identity statements relating two definite descriptions, as in the famous ‘The evening star is the morning star’. Strictly speaking this should not make sense, since the two types ‘Evening star’ and ‘Morning star’ differ. What is reasonable however is to consider these types as produced by dependent sum,

$$\sum_{x:Star} Shine\ brightly\ in\ morning(x).$$

Then while ‘the morning star’ is really a star, a warrant that it shines brightly in the morning, along with a guarantee of its uniqueness in this respect, a projection on the first component lands us in the type $Star$. The evening star is treated similarly, and now the two celestial bodies can be compared under the identity criteria for stars (as understood at the time).

3 Structure of

If we agree with the analysis of the previous section, then to be able to say ‘The structure of A’ by *the introduction* we must already have (a) formed a type ‘Structure of A’, and (b) established that it is contractible.

It seems to me that the only plausible candidate for ‘Structure of A’ is the type

$$Structure(A) \equiv \sum_{X:\mathcal{U}} f : Equiv(A, X),$$

where $\mathcal{U}$ is the type universe of small types, sometimes written $Type$. Elements of this type are types equipped with an equivalence with A. What is required now is to establish the contractibility of this type. Intuitively this should be clear as contraction can take place, as it were, along the given equivalences. But, of course, a proof in HoTT requires use of its technical apparatus which I will briefly sketch.

Straight off we have an element of that type, namely (A, Id_A). Then to establish $isContr(Structure(A))$ we also need for every B and $f : Equiv(A, B)$ a canonical way to identify (A, Id_A) and (B, f). What such an identity amounts to in the case of a dependent sum is a path in the base type, that is one in $\mathcal{U}$ between A and B, and a path over this one in the total space of equivalences to A. For the former we use the path that the univalence equivalence makes correspond to f. The effect of transporting $Id_A : Equiv(A, A)$ in the total space will then be $f \circ (Id_A) = f : Equiv(A, B)$. Let us call this process of identification p.

[We could also work with an equivalent type: $Structure(A) \equiv \sum_{X:\mathcal{U}}(A = X)$. Then Lemma 3.11.8 of the HoTT book gives us contractibility.]

4 The Structure of A

Now we are in a position to form the term 'the structure of A'. In full glory it is

$$the(Structure(A), (A, id_A), p) \equiv (A, id_A) : Structure(A).$$

Dropping p, we find that 'the structure of A' is (A, id_A). Notice that the component id_A is playing a role here. It might seem that we could say that 'the structure of A' is A itself, but it would be easy then to view it as potentially possessing nontrivial autoequivalences. We should note that, having constructed the type A, were we to construct an element $g : Equiv(A, A)$ which is not identical to id_A, then we could equally use (A, g) to witness the contractibility of $Structure(A)$. This would require a modification to p, but to the extent that this component is not mentioned, we might equally well say that (A, g) is 'the structure of A', or indeed any $(B, f) : Structure(A)$. Any such element has trivial self-identity type, $Id_{Structure(A)}((B, f), (B, f))$. An element of the type is an entity structured as A as witnessed by f.

Earlier I mentioned another approach to 'the' as the result of a function, as in 'the colour of my front door'. Here we might think there is a function from the type of types, $\mathcal{U}$, to some type of structures, *Structure*. As I explained there, this approach is reducible to the first approach, here via the dependent sum 'Structure which is (equivalent to) the structure of A'. The evident choice for *Structure* is the universe of types, $\mathcal{U}$, in which case we have the same solution as above, and the function 'the structure of' is found to be the identity function, $id_{\mathcal{U}} : \mathcal{U} \to \mathcal{U}$.

Now what does it mean to say that A and B have the same structure? Well one might expect that it means to indicate an identity between two elements 'the structure of A' and 'the structure of B'. However, strictly as in the case of the morning star and the evening star, they are elements of different types and so not to be compared. Like that example, we could project to the type that the dependent types are depending upon, here the universe $\mathcal{U}$. Then the identity of the elements amounts to an identity between types A and B in $\mathcal{U}$, or in other words equivalence between the types.

Alternatively we might define a $\mathcal{U}$-dependent type 'X has the same structure as A' $\equiv Equiv(A, X)$. Then consider by λ-abstraction the term $\lambda X.Equiv(A, X)$, which in words we might say designates 'has the same structure as A'. Now, 'has the same structure as B' is an element of the same type, and we can ask for their identity type. This can easily be shown to be equivalent to $Equiv(A, B)$.

5 Places in a Structure

Structuralist philosophers of mathematics have sometimes referred to 'places' or 'positions' in a structure. This is to indicate elements in what results from a process which abstracts away from different presentations of the 'same structure'. Let us see what it is possible to express within HoTT. Well, 'places in the structure of A' suggests that we form a type which depends on $Structure(A)$. There doesn't appear to be much choice here other than

$$(X, f) : Structure(A) \vdash PlacesIn(X, f) \equiv X : Type.$$

It would be very natural then to form the dependent product to allow the collection of coherent choices of element of A along with their corresponding elements according to given equivalences:

$$\prod_{(X,f):Structure(A)} PlacesIn(X, f).$$

We might pronounce this 'Places in A-structured types'. This type can easily be shown to be equivalent to A, since a choice $a : A$ determines elements $f(a)$ in structurally similar types.

6 Types Equipped with Structure

Of course, we don't just talk about plain types, but also about monoids, groups, vector spaces, etc. Consider one of the simplest cases, the semigroup structure. This merely requires that there be an associative binary multiplication on the type. Following the HoTT book's definition 2.14.1,

$$SemigroupStr(A) :\equiv \sum_{(m:A\to A\to A)} \prod_{(x,y,z:A)} m(x, m(y, z)) = m(m(x, y), z).$$

Now a semigroup is a type together with such a structure:

$$Semigroup :\equiv \sum_{A:\mathcal{U}} SemigroupStr(A).$$

Then for a particular $(A, m, a) : Semigroup$, where a is a proof of the associativity of m, we can define

$$Str(A, m, a) :\equiv \sum_{(X,y,z):Semigroup} f : Equiv_{Semigroup}((A, m, a), (X, y, z)),$$

where $Equiv_{Semigroup}$ requires of an element that it is an equivalence between carrier types and that it transports the semigroup structure correctly. Once again this results in a contractible type as witnessed by $(A, m, a, id_{(A,m,a)})$, which we may then call 'the structure of the semigroup A'. Places in (A, m, a)-structured semigroups will again amount to A.

7 Conclusion

One conclusion to draw from this note is that from the perspective of HoTT, little is gained by explicit use of the word 'structure' in the sense of 'the structure of A.' Types and structured types in HoTT simply *are* structures that do not need to be abstracted from an underlying set-like entity. As Mike Shulman (*forthcoming*) so aptly puts it, HoTT is a "synthetic theory of structures." The proper treatment of structure comes along for free and needs not be explicitly mentioned.

If this counts as a useful yet negative result, on the way to it we have seen something more positive. First, the analysis of the word *the* in terms of its introduction rule shows that HoTT has something to teach us about the classic philosophical topic of definite descriptions. We have seen that it provides a rationale for mathematicians' use of a generalized 'the' in situations where it appears that they might be referring to more than one entity.

Second, this analysis of 'the' employed a principle that may prove of lasting importance:

> (*Treat all types evenly*) Any time we have a construction which traditionally has been taken to apply only to sets or to propositions, then since in HoTT these form just a certain kind of type, we should look to see whether the construction makes sense for all types.

Further examples are not hard to find. If we generally take modal operators, such as 'it is necessarily the case that...,' to apply only to propositions, we should look to see whether there is anything to prevent a more general construction applying to all types 'necessary A'? This I shall consider in future work.

Acknowledgements: I would like to thank Mike Shulman and Urs Schreiber for their very helpful comments and suggestions.

8 Bibliography

Awodey, S. 2014. *Structuralism, Invariance, and Univalence*, (pdf)

Henriques, A. 2010. Comment on 'What's a groupoid? What's a good example of a groupoid?' (http://mathoverflow.net/a/1161)

Shulman, M. *forthcoming Homotopy Type Theory: A synthetic approach to higher equalities*, (pdf)

Univalent Foundations Program 2014, *Homotopy Type Theory*, (book)