Towards the syntax and semantics of higher dimensional type theory

Thorsten Altenkirch

May 25, 2018

The semantics of type theory is usually done on set level. An example is our definition of the intrinsic syntax of type theory [1] as a quotient inductive-inductive type (QIIT). Here we use a QIIT to construct the initial category with families (CwF) with a base family and Π-types. One fundamental problem of this construction is that we cannot interpret the syntax in the obvious semantic model that is sets. The reason is that **Set** is not a set but a groupoid and since we had to set-truncate the syntax we can only eliminate into a set. In our previous work we sidestepped the issue by using a inductively defined universe which is a set. However, this is clearly unsatisfying since we would like to interpret into a univalent universe. It seems strange that we cannot do this because all the required equations hold strictly in the intended **Set** model. Hence we would like to solve this coherence problem.

Another situation where a similar problem arises is the idea to use a directed type theory to specify higher inductive-inductive types as contexts in a theory of codes [3]. That is for example natural numbers can be specified as the context $X : U, z : X, s : X \to X$ in this system. However, since the syntax is again set-truncated we cannot hope to evaluate contexts to the proper higher inductive types.

We propose to develop a notion of a higher model of type theory within 2-level type theory [5] and use HIITs to construct a syntax for such a type theory. We conjecture that this syntax while not set-truncated is still a set but provides a more powerful elimination principle which addresses the problems described above.

A normal (set-level) CwF can be defined as follows.

- A category Con of contexts (and substitutions),
- A presheaf $\mathrm{Ty} : \mathrm{Con}^{\mathrm{op}} \to \mathbf{Set}$ of types,
- A presheaf $\mathrm{Tm} : (\int \mathrm{Ty})^{\mathrm{op}} \to \mathbf{Set}$ to model terms,
- A terminal object $\bullet$ in Con to denote the empty context,
- For any $A : \mathbf{Ty}(\Gamma)$, the presheaf

$$\Delta \mapsto \Sigma f : \mathbf{Con}(\Delta, \Gamma).A[f]$$

is representable.

Furthermore we usually assume a base type, a base family and we add type formers like Π-types by stipulating a natural transformation

$$\Pi_\Gamma : (\Sigma A : \mathrm{Ty}\Gamma, B : Ty(\Gamma, A)) \to \mathrm{Ty}\Gamma$$

and a natural (fibred) equivalence

$$\lambda_{\Gamma,A,B} : \mathrm{Tm}\,((\Gamma, A), B) \to \mathrm{Tm}(\Gamma, \Pi(A, B))$$

We define a ∞-CwF by doing the following substitutions:

- We replace category by $(\infty, 1)$-category,
- we observe that **Type** is an $(\infty, 1)$-category and replace set-valued presheaves by type valued presheaves,
- We define categories of elements for type-valued presheaves

What is a $(\infty, 1)$-category in HoTT? In [5] we explained how semi-simplicial types can be represented in 2-level HoTT and more general limits over an inverse category. Adding the Segal-condition (spines are equivalent to the simplex) which is just an equivalence, hence a proposition, we obtain $(\infty, 1)$-semicategories. How to go from semicategories to categories? Clearly, Δ is not inverse hence we cannot apply our construction. However, [2] show that univalence can be added as another property and it implies the existence of degeneracies. But there is another problem: a univalent category cannot have a set of objects and hence cannot be decidable, which is a property we would like to preserve. Luckily there is another option: in [4] a solution in form of a homotopy category (equivalences are labelled) which enables us to define simplicial sets as a type is presented. This seems to be a good base to define non-univalent $(\infty, 1)$-categories in HoTT.

Once the notion of a category is fixed the remaining components of a CwF can be defined. We need to establish that **Type** is a $(\infty, 1)$-category, morphisms between $(\infty, 1)$-categories which just correspond to a morphism between the corresponding strict presheaf categories. We need to define the category of elements in this setting, which seems fairly straightforward. The required morphisms are just 1-morphisms and being an equivalence is a proposition.

To define the syntax we need to define a family of HIT's indexed by the level. Each HIT gives rise to a $(\infty, 1)$-category and their colimit is the intended structure — we need to show in particular that this is a ∞-CwF with the required properties.

References

[1] Thorsten Altenkirch and Ambrus Kaposi. Type theory in type theory using quotient inductive types. *ACM SIGPLAN Notices*, 51(1):18–29, 2016.

[2] Paolo Capriotti and Nicolai Kraus. Univalent higher categories via complete semi-segal types. *Proceedings of the ACM on Programming Languages*, 2(POPL'18):44:1–44:29, dec 2017. Full version available at https://arxiv.org/abs/1707.03693.

[3] Ambrus Kaposi and Andras Kovacs. Codes for quotient inductive inductive types. https://akaposi.github.io/qiit.pdf, 2017.

[4] Nicolai Kraus and Christian Sattler. Space-valued diagrams, type-theoretically (extended abstract). *ArXiv e-prints*, 2017.

[5] Nicolai Kraus Thorsten Altenkirch, Paolo Capriotti. Extending homotopy type theory with strict equality. In *CSL*, 2016.