On the Role of Semisimplicial Types

Nicolai Kraus

University of Nottingham

Abstract

Constructing semisimplicial types is a well-known open problem in homotopy type theory. I explain why I believe that this problem is highly important. This talk proposal is based on several papers that are already available as well as on work in progress.

What are semisimplicial types? Let us write $\mathcal{U}$ for a universe in MLTT/HoTT. A *semisimplicial type* restricted to level 2 is a tuple (A_0, A_1, A_2) of the following types, where uncurrying is done implicitly:

$$\begin{aligned} &A_0 : \mathcal{U} \\ &A_1 : A_0 \to A_0 \to \mathcal{U} \\ &A_2 : (x, y, z : A_0) \to A_1(x, y) \to A_1(y, z) \to A_1(x, z) \to \mathcal{U} \end{aligned} \tag{1}$$

We can interpret A_0 as a type of points, A_1 as a type of (directed) lines between two given points, and A_3 as the type of fillers a triangle. On the next level, we would add a type family A_3 indexed over four points, six lines, and four triangle fillers which form a tetrahedron, and so on.[1]

Can we define semisimplicial types in HoTT? It is unknown whether there is a type family $F : \mathbb{N} \to \mathcal{U}_1$ such that $F(n)$ encodes the type of tuples $(A_0, \ldots, A_n)$ in any suitable way in "book HoTT" (the type theory developed in [Uni13a]). This is a major open problem in homotopy type theory, known as the problem of **defining semisimplicial types** [Uni13b]. It is easy to write down the definition if we assume UIP, which we do not in HoTT.

The problem has been considered so significant that other type theories which allow solutions have been suggested. The first is Voevodsky's *homotopy type system (HTS)* which enables us to reason about strict equality. The *two-level type theories (2LTTs)* as presented by Altenkirch, Capriotti and myself [ACK16] or Annenkov, Capriotti and myself [ACK17] are versions of HTS which offer some choices. Depending on these choices, one can get a 2LTT that is conservative over "book HoTT" [Cap16], or one in which semisimplicial types can be defined, but it is open whether both features can be combined. A form of 2LTT has been used by Boulier and Tabareau to define a model structure on the universe [BT17]. Another alternative to HTS is the *logic-enriched type theory* by Part and Lou [PL15]. As far as I know, a definition of semisimplicial types is also possible in the *computational higher type theory* of Angiuli, Favonia, Harper, and Wilson (see [AHW17] and related papers). I do not know what the status of semisimplicial types in other type theories is.

Why is this problem interesting? Voevodsky's original motivation for considering semisimplicial types was, I believe, their expected usefulness for type-theoretic versions of higher categorical structures. Such structures are very natural from a type-theoretic point of view, since types carry the structure of ∞-groupoids and universes the one of $(\infty, 1)$-categories. It is an open problem how to define $(\infty, 1)$-categories in "book HoTT". In settings where semisimplicial types are definable, *complete semi-Segal types* seem to be a well-behaved concept, as suggested by the work of Capriotti and myself [CK18]. More precisely, we define a *complete semi-Segal n-type* to be a semisimplicial type $(A_0, A_1, \ldots, A_{n+2})$ with three conditions: first, the *Segal condition*,

[1] Remark: This is an approach to encode type-valued presheaves over the category Δ_+ (the category of finite nonzero ordinals and strictly increasing functions) without having to talk about functor laws; it is inspired by the Reedy model structure for functor categories.

i.e. every inner horn has a contractible type of fillers; second, *completeness*, i.e. there is exactly one line which behaves like an isomorphism for every point; and third, a truncation condition ensuring that A_1 is a family of $(n-1)$-types. We then show that the complete semi-Segal 1-types correspond to the established notion of univalent 1-categories, and similar statements hold for 0 and 2. Having a definition of non-restricted semisimplicial types then allows us to lift the truncation condition to get a possible definition of $(\infty, 1)$-categories in type theory.

There are other constructions which become possible (or should be expected to become possible) if we have semisimplicial types at hand. Related to the above is the work by Sattler and myself: we can define types of diagrams over many different index categories (not just Δ_+), and we can show that these definitions are well-behaved [KS17].

A further very important question is whether homotopy type theory can serve as its own meta-theory, which one would expect from a foundation of mathematics. Shulman has discussed this in detail and explained that, if we can encode type theory in type theory, we also get a construction of semisimplicial types [Shu14]. It is reasonable to conjecture the opposite direction as well. In [AK16], Altenkirch and Kaposi present type theory in type theory, but with an explicit set-truncation which is problematic for homotopy type theory. I have hope that semisimplicial types make is possible to encode all required coherences such that the explicit truncation becomes unnecessary.

Do semisimplicial types also matter for synthetic homotopy theory? The actual question that I want to address is this this: Say we are interested in homotopy type theory but not so much in ∞-category theory, type theory eating itself, or higher coherence structures. For example, we could do *synthetic homotopy theory*, an important subfield of homotopy type theory which consists of formalising known or new constructions from homotopy theory by translating them into the more axiomatic setting of type theory. In the very impressive existing work it has so far not been necessary to consider infinite coherence structures explicitly. I believe that this is partially the case because clever encodings of the relevant data have been used to avoid such higher structures. One example of what I mean by this could already be the notion of a *half-adjoint equivalence*, where one gives only one out of two equations on level 2; the slightly less clever way would be to use both equations on level 2, then two coherence equations on level 3, and so on, already generation an infinite (but in this case less complicated) tower. In any case, it is not clear that such encodings are possible in all situations that might turn up in HoTT.

One case where I expect that semisimplicial types play a role (although I have not yet fully worked it out) is the open problem recorded in [Uni13a, Exercise 8.2]. Given a set (0-truncated type) A, the question is whether the suspension $\mathsf{Susp}(A)$ is always 1-truncated. I conjecture that the question can be answered positively if semisimplicial types are available in the type theory (this is related to the forthcoming paper [KA18] which however only shows a weaker statement, namely that the second homotopy group of $\mathsf{Susp}(A+1)$ is trivial). Let us try to understand what the problem has to do with semisimplicial types. If we attempt to use the encode-decode method (see [Uni13a, Chp 2.12]) to characterise the path spaces of the said suspension, we quickly realise that any list of elements of A gives rise to a path. However, two lists can encode the same path if two occurrences of a single $a : A$ appear next to each other, and we would have to quotient them out. But this opens the door for coherence problems if there are multiple such pairs; removing one and then another should be equal to removing the second one and then the first, and so on. In short, we need to form a quotient with coherence conditions that are reminiscent of semisimplicial types. The arguments I am using to attack the problem make use of (a special case of) another of my results that depend on semisimplicial types, namely the fact that functions $\|A\|_{-1} \to B$ correspond to *coherently constant functions* $A \to B$ [Kra15a, Kra15b].

References

[ACK16] Thorsten Altenkirch, Paolo Capriotti, and Nicolai Kraus. Extending homotopy type theory with strict equality. *CSL'16*, 2016.

[ACK17] Danil Annenkov, Paolo Capriotti, and Nicolai Kraus. Two-level type theory and applications. 2017. arXiv:1705.03307.

[AHW17] Carlo Angiuli, Robert Harper, and Todd Wilson. Computational higher-dimensional type theory. *POPL'17*, 2017.

[AK16] Thorsten Altenkirch and Ambrus Kaposi. Type theory in type theory using quotient inductive types. *POPL'16*, 2016.

[BT17] Simon Boulier and Nicolas Tabareau. Model structure on the universe in a two level type theory. 2017. https://hal.archives-ouvertes.fr/hal-01579822.

[Cap16] Paolo Capriotti. *Models of Type Theory with Strict Equality.* PhD thesis, School of Computer Science, University of Nottingham, 2016.

[CK18] Paolo Capriotti and Nicolai Kraus. Univalent higher categories via complete semi-Segal types. *POPL'18*, 2018.

[KA18] Nicolai Kraus and Thorsten Altenkirch. Free higher groups in homotopy type theory. 2018. To appear.

[Kra15a] Nicolai Kraus. The general universal property of the propositional truncation. *TYPES'14 proceedings*, 2015.

[Kra15b] Nicolai Kraus. *Truncation Levels in Homotopy Type Theory.* PhD thesis, School of Computer Science, University of Nottingham, 2015.

[KS17] Nicolai Kraus and Christian Sattler. Space-valued diagrams, type-theoretically (extended abstract). 2017. arXiv:1704.04543.

[PL15] Fedor Part and Zhaohui Luo. Semi-simplicial types in logic-enriched homotopy type theory. 2015. arxiv:1506.04998.

[Shu14] Michael Shulman. Homotopy type theory should eat itself (but so far, it's too big to swallow), 2014. Blog post, `homotopytypetheory.org/2014/03/03/hott-should-eat-itself`.

[Uni13a] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics.* `http://homotopytypetheory.org/book/`, Institute for Advanced Study, 2013.

[Uni13b] The Univalent Foundations Program. Semi-simplicial types, 2013. Wiki page, `https://uf-ias-2012.wikispaces.com/Semi-simplicial+types`.