GEOMETRIC REALIZATION OF TRUNCATED SEMI-SIMPLICIAL SETS META-CONSTRUCTED WITHIN HOTT

GENKI SATO

In this talk, I talk about a construction analogous to geometric realization of truncated semi-simplicial sets, within HoTT. That is, let n be a meta-level natural number and X be a semi-simplicial 0-type. By geometric realization I mean the type $\mathrm{GR}_n(X)$ constructed by the iterated homotopy pushouts of the form

$$\begin{tikzcd} X_n \times \mathrm{GR}_{n-1}(\partial\Delta\,[n]) \arrow[r] \arrow[d] & X_n \arrow[d] \\ \mathrm{GR}_{n-1}(X) \arrow[r] & \mathrm{GR}_n(X). \end{tikzcd}$$

Here, I used semi-simplicial 0-type instead of truncated semi-simplicial type just for ease of notation: in order to omit the $(n-1)$-truncation functor applied to an n-truncated semi-simplicial type X. The construction GR_n applies to general n-truncated semi-simplicial types.

This construction is intuitive and looks easy at first sight, but turns out to be technically difficult. The difficulty is in defining the intuitively obvious vertical arrow $X_n \times \mathrm{GR}_{n-1}(\partial\Delta\,[n]) \to \mathrm{GR}_{n-1}(X)$. For the rigorous construction of this arrow, we need the 1-functoriality of GR_{n-1}, which allows us to carry the morphism $\partial\Delta\,[n] \to X$ corresponding to each element in X_n to a map $\mathrm{GR}_{n-1}(\partial\Delta\,[n]) \to \mathrm{GR}_{n-1}(X)$. For the 1-functoriality of GR_{n-1}, we need the 2-functoriality, by which I mean the analogue of a map between the set of 2-simplices in quasi-categories, of GR_{n-2}, and therefore need the 3-functoriality of GR_{n-3}, and so on. Even before proving this, we need a precise language for expressing and defining the higher functoriality of geometric realizations, or more elementarily, of homotopy pushouts.

The main point of this talk is to build the language for this purpose. What we need is the concept of higher-dimensional diagrams in HoTT, and therefore that of higher-dimensional paths of various shapes, which I have constructed meta-parametrically in the shape, following and generalizing Dan Licata and Guillaum Bruneries's method of defining the cubical language outside cubical type theory, back in Martin-Löf type theory [1].

Let K be a shape like a polyhedron and $C(K)$ be the face category of K: the category with its objects the faces of K and its morphisms the inclusions between them. However we need to note that K is in fact not an actual mathematical existence; it is $C(K)$ that we actually meta-parametrize our construction in. In fact, $C(K)$ may be any direct category, i.e. any category admitting a functor bijective in objects to a partially ordered set. Although we may not always construct K as a polyhedron, but we can see the nerve $N(C(K)) \in \mathrm{Set}_{\mathbf{\Delta}}$ as the "barycentric subdivision" of the "shape" K.

We may construct, then, a type family modeling the fibration

$$X^K \to X^{\partial K},$$

by taking the homotopy type theoretic homotopy fibers of the "diagonal map"

$$X \to X^{\partial K}.$$

Here X is any type (seen as a ∞-groupoid), $C(K)$ is assumed to have a terminal object, and ∂K corresponds to $C(K)$ minus the terminal object. Gathering X^{K_c} for each $c \in \mathrm{Ob}(C(K))$ and matching variables, where K_c corresponds to the slice $C(K)/c$, yields the definition of $X^{\partial K}$. Strictly speaking, we may think of $X^{\partial K}$ as a context beginning with X type, or the iterated Σ-type of that context minus the first entry X type.

With this language, we may define simplices and cubes in types, by taking K to be simplices and cubes and $C(K)$ to be their face category. Furthermore, by modifying the construction of the homotopy coherent nerves of simplicially enriched categories with caution to the fact that horizontal compositions of positive-dimensional homotopies is only homotopy-associative in HoTT, we obtain the definition of "n-simplices in the quasi-category of types", or more generally, of diagrams with the shape $\Delta[k_1] \times \Delta[k_2] \times \cdots \times \Delta[k_n]$.

Then the k-functoriality of homotopy pushout, for example, may be stated as the extension property of diagrams with the shape

$$(\Delta[1] \times \Delta[0] \cup \Delta[0] \times \Delta[1]) \times \Delta[k].$$

This reduces, up to the universality of homotopy pushout, to some extensions of corresponding path space (which is compared to finding a nice section to the restriction $X^L \twoheadrightarrow X^K$ for appropriate inclusions $K \hookrightarrow L$). A sufficient condition for the possibility of such extensions may be stated as a purely combinatorial property of the "inclusion functor" $C(K) \to C(L)$. By carefully examining the combinatorics of the related categories, we may prove the ∞-functoriality of homotopy pushout. Also, I believe this yields (though I need to re-check the proof of) the ∞-functoriality of geometric realization, thereby completing the construction of geometric realization.

As an application, I currently conjecture that geometric realization preserves many homotopy-theoretic properties; for example I suspect that GR_n preserves the k-th homotopy group for $k < n$. I believe that, if such is proven, then combining this with the set theory in HoTT, we get a completely formulated proof of provability of certain ZFC-provable homotopy-theoretic provabiltiy in HoTT equipped with the axiom of choice; for instance, computability of ZFC-computable homotopy groups of spheres in HoTT+AC.

Finally, we again note that the constructions here are meta-theoretic. It is likely that these constructions cannot be made internally parametric with respect to the dimension or the shape, for the same reason why we conjecture that the concept of semi-simplicial types is not internally definable in HoTT with no extensions. The internalization of these constructions within extended HoTTs may be a matter of interest.

References

[1] Daniel R. Licata and Guillaume Brunerie. 2015. A Cubical Approach to Synthetic Homotopy Theory. In Proceedings of the 2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS) (LICS '15). IEEE Computer Society, Washington, DC, USA, 92-103.