Kan Operations for n-Truncated Types

KANG Rongji

September 16, 2023

Abstract

In this note, we introduce a new kind of Kan operations in cubical type theory, specifically defined for n-truncated types. They provide an efficient way to construct cubes within truncated types.

Contents

Notation We use a syntax style based on `Cubical Agda` to present our results. Pay special attention that (possibly non-fibrant) extension types are expressed as

$$(\vec{\imath} : \mathbb{I}^n|_{\psi}) \to X\ \vec{\imath}\ \{\phi \mapsto x\}$$

Here $\vec{\imath}$ should be understood as an n-tuple of interval variables, and ψ and ϕ are cofibrations. The terms of this type are exactly those of $X\ \vec{\imath}$ that are defined only when judgement ψ `true` holds, and whenever ϕ `true` also holds it should be judgementally equal to x. We use $\partial\mathbb{I}^n$ to denote the cofibration corresponding to the boundary of an n-cube.

Throughout this note, we prefer the notion of h-levels, instead of n-truncatedness. Remember that being of h-level n is synonymous with being $(n-2)$-truncated.

1 Introduction

This note aims to construct a Kan operation, for each h-level n, in cubical type theory:

$$
\begin{aligned}
&\mathsf{extend}_n : \\
&\quad (X : \mathbb{I}^n \to \mathsf{Type}) \\
&\quad (h : (\vec{\imath} : \mathbb{I}^n) \to \mathsf{isOfHLevel}_n\ (X\ \vec{\imath})) \\
&\quad (\phi : \mathsf{Cofibration}) \\
&\quad (x : (\vec{\imath} : \mathbb{I}^n|_{\phi \cup \partial\mathbb{I}^n}) \to X\ \vec{\imath}) \\
&\quad (\vec{\imath} : \mathbb{I}^n) \to X\ \vec{\imath}\ \{(\phi \cup \partial\mathbb{I}^n) \mapsto x\}
\end{aligned}
$$

It allows us to unconditionally extend any partial term defined within part of a context, provided it is fully defined over the boundary of an n-cube. This operation is powerful, and, not hard to show conversely, it serves as a characterization of h-levels. Thus, we arrive at the following theorem:

Theorem 1. *A type family X is of h-levels n if and only if the operation extend_n could be defined for it.*

In the case of contractible types, namely when $n = 0$, it asserts all partial terms can be extended. It's no surprise, as contractibility means a certain triviality in homotopy-theoretic setting. This special case has already been established in Section 5.1 of [1], referred as contr. Our theorem can be viewed as a generalization of it.

2 h-Levels

In homotopy theory, the concept of being h-level n coincides with what homotopy theorists refer to as being $(n-2)$-truncated. It is often defined inductively, as originally presented in in the book [2], with the base case, $n = 0$, corresponds to contractibility.

$$\mathsf{isOfHLevel}_0\ X = \mathsf{isContr}\ X = \Sigma_{(a:X)}((x : X) \to \mathsf{Path}\ a\ x)$$

In general, a type has h-level $n+1$ precisely when its path types all have h-level n. This recursive definition establishes a hierarchy of h-levels.

$$\mathsf{isOfHLevel}_{n+1}\ X = (a\ b : X) \to \mathsf{isOfHLevel}_n\ (\mathsf{Path}\ a\ b)$$

Unfortunately, the inductive pattern usually deviates at $n = 1$, particularly in the case of h-propositions. Here, it is defined differently but equivalent to the previous definition.

$$\mathsf{isOfHLevel}_1\ X = \mathsf{isProp}\ X = (a\ b : X) \to \mathsf{Path}\ a\ b$$

This definition inherently relies on induction, which can sometimes cause trouble to extract information for a specific n directly. There are alternative, less-inductive definitions available, such as the assertion that any function from n-spheres S^n is null-homotopic:

$$(f : \mathsf{S}^n \to X) \to \Sigma_{a:X}((x : \mathsf{S}^n) \to \mathsf{Path}\ a\ (f\ x))$$

Nevertheless, using S^n in homotopy-theoretic arguments can still put forward its own kinds of difficulty.

Our proposal is to use extend_n instead, as its existence, demonstrated in Theorem 1, serves as a characterization of h-level. Notably, there is similarity between our operation and the definition using spheres. However, our operation offers better flexibility, thanks to the additional cofibration ϕ, and could be applied more directly within the framework of cubical type theory. Later we will give a simple application.

To proceed with our construction, we need a straightforward lemma.

Lemma 2. *Given a type family $P : \mathbb{I} \to \mathsf{Types}$ of h-level $n+1$, its associated heterogeneous path types all have h-level n.*

Proof. We need to show that for any $a : P\ 0$ and $b : P\ 1$, the type $\mathsf{PathP}\ P\ a\ b$ is of h-level n. First transport backward P to find a heterogeneous path $p : \mathsf{PathP}\ P\ b'\ b$ for some $b' : P\ 0$. Then transport forward $\lambda\ i \mapsto \mathsf{isOfHLevel}_n\ (\mathsf{PathP}\ P\ a\ (p\ i))$ to finish the proof. $\square$

3 The Construction

Recall that the base case extend_0 has been established in [1]. We proceed with an induction on h-levels, assuming the existence of extend_n. For convenience, we decompose the variable of an $(n+1)$-cube into an n-cube one and a single interval one:

$$(\vec{\imath}, j) : \mathbb{I}^{n+1}$$

It's a bit amusing that the whole proof is in some sense just an application of such a trivial decomposition. We begin by fixing the family X, the witness h of h-level and the cofibration ϕ. The crucial observation is that, what we are seeking for:

$$\begin{aligned}
&\mathsf{extend}_{n+1} : \\
&\quad (x : ((\vec{\imath}, j) : \mathbb{I}^{n+1}|_{\phi \cup \partial\mathbb{I}^{n+1}}) \to X\ \vec{\imath}\ j) \\
&\quad ((\vec{\imath}, j) : \mathbb{I}^{n+1}) \\
&\quad \to X\ \vec{\imath}\ j\ \ \{(\phi \cup \partial\mathbb{I}^{n+1}) \mapsto x\ \vec{\imath}\ j\}
\end{aligned}$$

is almost the same as the following operation:

$$\begin{aligned}
&\mathsf{extend}_n\mathsf{PathP} : \\
&\quad (x : ((\vec{\imath}, j) : \mathbb{I}^{n+1}|_{\phi \cup \partial \mathbb{I}^{n+1}}) \to X\ \vec{\imath}\ j) \\
&\quad (\vec{\imath} : \mathbb{I}^n) \\
&\quad \to \mathsf{PathP}\ (\lambda\ j \mapsto X\ \vec{\imath}\ j)\ (x\ \vec{\imath}\ 0)\ (x\ \vec{\imath}\ 1) \\
&\qquad\quad \{(\phi \cup \partial \mathbb{I}^n) \mapsto (\lambda\ j \mapsto x\ \vec{\imath}\ j)\}
\end{aligned}$$

The only difference lies in how they handle the variable j. One puts it inside the extension type, while the other places it outside. Though their type signatures cannot be identified judgementally, they are deducible from one another.

Using Lemma 2, we can show that for each $\vec{\imath} : \mathbb{I}^n$, the heterogeneous path type associated to $X\ \vec{\imath}$ is of h-levels n:

$$(a : X\ \vec{\imath}\ 0)\ (b : X\ \vec{\imath}\ 1) \to \mathsf{isOfHLevel}\ (\mathsf{PathP}\ (\lambda\ j \mapsto X\ \vec{\imath}\ j)\ a\ b)$$

The induction strategy is now clear. We apply first the given extend_n on PathP to obtain an $\mathsf{extend}_n\mathsf{PathP}$. Then we resort to our earlier observation to construct extend_{n+1}.

4 Application: Filling Cubes in n-Types

As a quick application, it can be shown immediately that for a type X of h-level n, from any given boundary the m-cubes ($m \geq n$) can be filled in X.

$$\mathsf{fillCube} : (x : (\vec{\imath} : \mathbb{I}^m|_{\partial \mathbb{I}^m}) \to X)\ (\vec{\imath} : \mathbb{I}^m) \to X\ \{\partial \mathbb{I}^m \mapsto x\ \vec{\imath}\}$$

Let's write $m = n + k$ for some non-negative k. We separate the m-cube variable into two parts, where $\vec{\imath} : \mathbb{I}^n$ and $\vec{\jmath} : \mathbb{I}^k$:

$$(\vec{\imath}, \vec{\jmath}) : \mathbb{I}^m$$

Next we feed $\vec{\imath}$ into extend_n while put $\vec{\jmath}$ in the context. The resulting output is the desired filling. A relative version can be similarly established.

5 Semantics

In classical homotopy theory, a fibration $p : X \to S$ is $(n-2)$-truncated, if and only if the following extension-lifting problem, whatever the cofibration $i : A \to B$ is chosen, always admits a solution:

$$\begin{tikzcd}
A \times \mathbb{I}^n \cup B \times \partial \mathbb{I}^n \arrow[r] \arrow[d, hook] & X \arrow[d, "p"] \\
B \times \mathbb{I}^n \arrow[r] \arrow[ur, dotted] & S
\end{tikzcd}$$

Our Theorem 1 certainly aligns with this characterization. However, the existence of extend_n is a bit stronger, since it yields a specific result that remains stable under arbitrary substitution. This is the consequence of working in a more constructive setting.

6 Final Remarks

The construction of extend_n only relies on basic geometric facts about cofibration and extension types. So it is reasonable to claim that most cubical type theories should be able to realize it.

If we define the property of h-level n as the existence of extend_n, some basic facts about h-levels could be deduced more straightforwardly. It shows the effectiveness of cubical methods in handling homotopy-theoretic concepts. The use of extension types is crucial if we want extend_n to be operations parametrized by internal natural number $n : \mathbb{N}$. This fact highlights the value of extension types.

For small values of n, a formalization of extend_n has been done in `Cubical Agda` [**?**]. The bad news is, it does not support extension type yet. We can only regard Theorem 1 as a meta-theorem, and a full formalization appears to be impossible. Nonetheless, it seems not hard to construct a macro that automatically generates extend_n for any given external natural number $n \in \mathbb{N}$.

References

[1] Cyril Cohen, Thierry Coquand, Simon Huber, and Anders Mörtberg, *Cubical type theory: a constructive interpretation of the univalence axiom*, arXiv preprint arXiv:1611.02108 (2016).

[2] The Univalent Foundations Program, *Homotopy type theory: Univalent foundations of mathematics*, `https://homotopytypetheory.org/book`, Institute for Advanced Study, 2013.