Finitary Higher Inductive Types in the Groupoid Model

Peter Dybjer and Hugo Moeneclaey

Chalmers and ENS Cachan

Workshop on Foundations
for the Practical Formalization of Mathematics
Nantes, 26-27 April, 2017

What is a higher inductive type?

- In ordinary Martin-Löf type theory

$$a =_A a'$$

has *one* constructor $\mathrm{refl} : a =_A a$.

- In Homotopy Type Theory higher inductive types (hits) are types A where we can have other constructors as well, for all the iterated identity types:

$$
\begin{gathered}
a =_A a' \\
p =_{a =_A a'} p' \\
\theta =_{p =_{a =_A a'} p'} \theta' \\
\vdots
\end{gathered}
$$

Bauer, Lumsdaine, Shulman, Warren 2011.

Higher inductive types of level n

Terminology:

- point constructor for A (level 0)
- path constructor for $a =_A a'$ (level 1)
- surface constructor for $p =_{a =_A a'} p'$ (level 2)
- etc

n-hits only have constructors of level $\leq n$.

1-hits in the HoTT-book

General examples:

- propositional truncation
- pushout

Homotopical examples:

- interval
- circle
- suspension

2-hits

General examples:

- 0-truncation
- set-quotient

Homotopical examples:

- 2-sphere
- torus

Computer science example:

- patch theories (Angiuli, Harper, Licata, Morehouse, 2014)

From the HoTT-book

In this book we do not attempt to give a general formulation of what constitutes a "higher inductive definition" and how to extract the elimination rule from such a definition - indeed, this is a subtle question and the subject of current research. Instead we will rely on some general informal discussion and numerous examples.

Some questions

- What is a good definition of a higher inductive type, that is, what do the types of their constructors look like in general?
- What are their associated elimination and equality rules?
- How do we show the consistency of a general theory of higher inductive types?
- How do we get a "computational interpretation"?
- What is the foundational status of higher inductive types? What is their relation to Martin-Löf's meaning explanations?
- Can we reduce the meaning of higher inductive types to the standard inductive or inductive-recursive types?

Higher-dimensional, univalent type theory

A *reinterpretation* of *intensional* type theory

- type $=$ weak ∞-groupoid (Kan cubical set)
- new rules are validated, e g the *univalence* axiom and *higher inductive types*
- constructivity is maintained because Kan cubical set model can be formulated in constructive metatheory (*extensional* type theory) itself justified by Martin-Löf's (1979) *standard* meaning explanations. Cf work in progress by Bickford and Coquand on an implementation in NuPRL.

Type theory in the groupoid model

A *reinterpretation* of *intensional* type theory, Hofmann and Streicher (1993).

- type = groupoid $A = (A_0, A_1, A_2) = (A_0, A_1, =_{A_1(\_,\_)})$.
- new rules are validated, e g univalence axiom in first universe and *higher inductive types* of level 2.
- constructivity is maintained because groupoid model can be formulated in constructive metatheory (*extensional* type theory) itself justified by Martin-Löf's (1979) *standard* meaning explanations.

Type theory in the setoid model

A *reinterpretation* of *intensional* type theory

- type = setoid $A = (A_0, A_1) = (A_0, =_A)$.
- new rules are validated, e g *higher inductive types* of level 1, including quotient types and algebraic theories $\mathrm{T}_{\Sigma,E}$. Cf Basold, Geuvers, van der Weide (2017).
- constructivity is maintained because setoid model can be formulated in constructive metatheory (*extensional* type theory) itself justified by Martin-Löf's (1979) *standard* meaning explanations.

Quotient types

Let A be a type and R be a binary relation on A. Then A/R is the 1-hit with

$$\begin{aligned} \mathrm{c}_0 &: A \to A/R \\ \mathrm{c}_1 &: (x, y : A) \to R(x, y) \to \mathrm{c}_0(x) =_{A/R} \mathrm{c}_0(y) \end{aligned}$$

Notation: $[x] = \mathrm{c}_0(x)$

Quotient types in the setoid model

In the setoid model the points/elements are generated by the constructor

$$\mathrm{c}_{00} \quad : \quad A_0 \to (A/R)_0$$

and the paths/proofs of equality are generated by

$$\begin{aligned}
\mathrm{c}_{10} \quad &: \quad (x, y : A_0) \to (R(x, y))_0 \to \mathrm{c}_{00}(x) =_{A/R} \mathrm{c}_{00}(y) \\
\mathrm{c}_{01} \quad &: \quad (x, y : A_0) \to x =_A y \to \mathrm{c}_{00}(x) =_{A/R} \mathrm{c}_{00}(y) \\
\circ \quad &: \quad (x, y, z \in (A/R)_0) \to x =_{A/R} y \to y =_{A/R} z \to x =_{A/R} z \\
\mathrm{id} \quad &: \quad (x \in (A/R)_0) \to x =_{A/R} x \\
(-)^{-1} \quad &: \quad (x, y \in (A/R)_0) \to x =_{A/R} y \to y =_{A/R} x
\end{aligned}$$

Note that $(A/R)_0$ is an inductive type and $=_{A/R}$ is an inductive family which are instances of the general schema for inductive families of Dybjer (1991) and CiC.

Heterogenous identity

- If $x : A \vdash C(x)$, $a, a' : A$, and $p : a =_A a'$,then

$$c =_p^C c'$$

  denotes the heterogenous identity of $c : C(a)$ and $c' : C(a')$.
- If $f : (x : A) \to C(x)$, $a, a' : A$, then

$$\mathbf{apd}_f : (p : a =_A a') \to f(a) =_p^C f(a')$$

Both are definable from the rules for homogeneous identity types. (Should they perhaps be primitive?)

Elimination and equality rules for quotients

The elimination rule expresses how to define a function

$$f : (x : A/R) \to C(x)$$

by structural induction on the points of A/R, such that the function preserves $=_{A/R}$.

$$\begin{aligned} f(\mathrm{c}_0(x)) &= \tilde{\mathrm{c}_0}(x) \\ \mathbf{apd}_f(\mathrm{c}_1(x,y,z)) &= \tilde{\mathrm{c}_1}(x,y,z) \end{aligned}$$

under the assumptions

$$\begin{aligned} \tilde{\mathrm{c}_0} &: (x : A) \to C(\mathrm{c}_0(x)) \\ \tilde{\mathrm{c}_1} &: (x,y : A) \to (z : R(x,y)) \to \tilde{\mathrm{c}_0}(x) =^{C}_{\mathrm{c}_1(x,y,z)} \tilde{\mathrm{c}_0}(y) \end{aligned}$$

General schema for 1-hits?

H is a hit with point constructors

$$\mathrm{c}_0 : ?$$

and path constructors

$$\mathrm{c}_1 : ?$$

What is the form of their types?

General schema for 1-hits?

H is a hit with point constructors

$$\mathrm{c}_0 : ?$$

and path constructors

$$\mathrm{c}_1 : ?$$

What is the form of their types? First try:

- the type of a point constructor has the form of a constructor for an inductive type H.
- the type of a path constructor has the form of a constructor for a binary inductive family $=_{\mathrm{H}}$ on H.

A schema for finitary 1-hits

We settle for the time being for a restricted version of hits:

- the type of a point constructor has the form of a constructor for a *finitary* inductive type H.
- the type of a path constructor has the form of a constructor for a *finitary* binary inductive family $=_{\mathrm{H}}$ on H. The indices in the type are *point constructor patterns*

Three reasons:

- Simpler semantics
- Simpler syntax, yet cover most (but not all) examples
- Clearly constructive (the schema for inductive families in Dybjer (1991) was perhaps too general)

The type of a point constructor

Finitely branching trees, with finitely many constructors

$$
\begin{aligned}
\mathrm{c}_0 \quad : \quad & (x_1 : A_1) \to \cdots \to (x_n : A_n(x_1, \ldots, x_{n-1})) \\
& \to \mathrm{H} \to \cdots \to \mathrm{H} \\
& \to \mathrm{H}
\end{aligned}
$$

A_i are arbitrary types. They may not depend on H.
This is also the schema for point constructors of the hit H.

A schema for path constructors

$$
\begin{aligned}
\mathrm{c}_1 \quad : \quad & (x_1 : A_1) \to \cdots \to (x_n : A_n(x_1, \ldots, x_{n-1})) \\
& \to (y_1 : \mathrm{H}) \to \cdots \to (y_{n'} : \mathrm{H}) \\
& \to p_1(x_i, y_j) =_{\mathrm{H}} q_1(x_i, y_j) \to \cdots \to p_m(x_i, y_j) =_{\mathrm{H}} q_m(x_i, y_j) \\
& \to p'(x_i, y_j) =_{\mathrm{H}} q'(x_i, y_j)
\end{aligned}
$$

where neither H nor $=_{\mathrm{H}}$ may appear in A_i and where $p_1, q_1, \ldots, p_m, q_m, p', q'$ are *point constructor patterns* built up by from variables x_i, y_j by point constructors c_0.

- 1-hits generalize $\mathrm{T}_{\Sigma,E}$ from algebraic specification theory, the initial term algebra for a signature Σ and a list of equations E.
- Note that although one may think that the set of points of H is defined before $=_{\mathrm{H}}$, a negative occurrence of H would generate a negative occurrence of $=_{\mathrm{H}}$ in the setoid interpretation of $=_{\mathrm{H}}$.

Simplified schema for 1-hits

A simplified form with only one side condition and one inductive premise:

$$
\begin{aligned}
\mathrm{c}_0 \quad &: \quad A_0 \to \mathrm{H} \to \mathrm{H} \\
\mathrm{c}_1 \quad &: \quad (x : A_1) \to (y : \mathrm{H}) \to p(x, y) =_{\mathrm{H}} q(x, y) \\
&\quad\;\; \to p'(x, y) =_{\mathrm{H}} q'(x, y)
\end{aligned}
$$

The Torus T^2 as a 2-hit

$$
\begin{array}{rcl}
\mathrm{base} & : & \mathrm{T}^2 \\
\mathrm{path}_1 & : & \mathrm{base} =_{\mathrm{T}^2} \mathrm{base} \\
\mathrm{path}_2 & : & \mathrm{base} =_{\mathrm{T}^2} \mathrm{base} \\
\mathrm{surf} & : & \mathrm{path}_1 \cdot \mathrm{path}_2 =_{\mathrm{base} =_{\mathrm{T}^2} \mathrm{base}} \mathrm{path}_2 \cdot \mathrm{path}_1
\end{array}
$$

Simplified schema for 2-hits

Simplified version:

$$
\begin{aligned}
\mathrm{c}_0 &: A_0 \to \mathrm{H} \to \mathrm{H} \\
\mathrm{c}_1 &: (x : A_1) \to (y : \mathrm{H}) \to p(x,y) =_{\mathrm{H}} q(x,y) \\
&\quad \to p_1(x,y) =_{\mathrm{H}} q_1(x,y) \\
\mathrm{c}_2 &: (x : A_2) \to (y : \mathrm{H}) \to (z : p_2(x,y) =_{\mathrm{H}} q_2(x,y)) \\
&\quad \to g_1(x,y,z) =_{p_3(x,y) =_{\mathrm{H}} q_3(x,y)} h_1(x,y,z) \\
&\quad \to g_2(x,y,z) =_{p_4(x,y) =_{\mathrm{H}} q_4(x,y)} h_2(x,y,z)
\end{aligned}
$$

Here p, q, p_i, q_i are *point constructor patterns* in the variables x, y and g_i, h_i are *path constructor patterns* in the variables x, y, z.

Point and path constructor patterns

Point constructor patterns

$$p ::= x \mid \mathrm{c}_0(a, p)$$

Path constructor patterns

$$g ::= z \mid \mathrm{c}_1(a, p, g) \mid g \circ g \mid \mathrm{id} \mid g^{-1}$$

(add $\mathbf{ap}_{\mathrm{c}_0}(p, g)$?)

Elimination rule for the simplified schema for hits

The elimination rule expresses how to define a function

$$f : (x : \mathrm{H}) \to C(x)$$

by structural induction on the points of H, such that the function preserves $=_{\mathrm{H}}$.

$$\begin{aligned} f(\mathrm{c}_0(x,y)) &= \tilde{\mathrm{c}}_0(x,y,f(y)) \\ \mathbf{apd}_f(\mathrm{c}_1(x,y,z)) &= \tilde{\mathrm{c}}_1(x,y,f(y),z,\mathbf{apd}_f(z)) \end{aligned}$$

under the assumptions

$$\begin{aligned} \tilde{\mathrm{c}}_0 &: (x : A_0) \to (y : \mathrm{H}) \to C(y) \to C(\mathrm{c}_0(x,y)) \\ \tilde{\mathrm{c}}_1 &: (x : A_1) \to (y : \mathrm{H}) \to (\tilde{y} : C(y)) \\ &\to (z : p =_{\mathrm{H}} q) \to \mathrm{T}_0(p) =^C_z \mathrm{T}_0(q) \to \mathrm{T}_0(p') =^C_{\mathrm{c}_1(x,y,z)} \mathrm{T}_0(q') \end{aligned}$$

where T_0 is a *lifting function* defined below.

Lifting

Lifting point constructor patterns:

$$\begin{aligned} \mathrm{T}_0(y) &= \tilde{y} \\ \mathrm{T}_0(\mathrm{c}_0(a,p)) &= \tilde{\mathrm{c}_0}(a,p,\mathrm{T}_0(p)) \end{aligned}$$

Lifting path constructor patterns:

$$\begin{aligned} \mathrm{T}_1(z) &= \tilde{z} \\ \mathrm{T}_1(\mathrm{c}_1(a,p,g)) &= \tilde{\mathrm{c}_1}(a,p,\mathrm{T}_0(p),g,\mathrm{T}_1(g)) \\ \mathrm{T}_1(g \circ g') &= \mathrm{T}_1(g) \circ' \mathrm{T}_1(g') \\ \mathrm{T}_1(\mathrm{id}) &= \mathrm{id} \\ \mathrm{T}_1(g^{-1}) &= \mathrm{T}_1(g)^{-1'} \end{aligned}$$

It follows that $\mathrm{T}_0(p) = f(p)$ and $\mathrm{T}_1(g) = \mathbf{ap}_f(g)$

Heterogeneous identity of level 2.

Let $a, a' : A$, $p, p' : a =_A a'$, $\theta : p =_{a =_A a'} p'$, $b : B(a)$, $b' : B(a')$, $q : b =^B_p b'$, $q' : b =^B_{p'} b'$ We write

$$q =^{b =^B_{-} b'}_{\theta} q'$$

for the heterogeneous identity of the heterogenous paths q, q'. Moroever,

$$q =^{b =^B_{-} b'}_{\mathrm{refl}(p)} q'$$

is judgmentally equal to $q =_{b =^B_p b'} q'$.

Functions preserve level 2 identities

If

$$f : (x : A) \to C(x)$$

then not only

$$\mathbf{apd}_f : (p : x =_A x') \to f(x) =^C_p f(x')$$

but also

$$\mathbf{apd}^2_f : (\theta : p =_{x =_A x'} p') \to \mathbf{apd}_f(p) =^{f(x) =^C_- f(x')}_\theta \mathbf{apd}_f(p')$$

Elimination and equality rules

We define $f : (x : \mathrm{H}) \to C(x)$ by

$$\begin{aligned} f(\mathrm{c}_0(a_1, b_1)) &= \tilde{\mathrm{c}_0}(a_1, b_1, f(b_1)) \\ \mathbf{apd}_f(\mathrm{c}_1(a_2, b_2, c_2)) &= \tilde{\mathrm{c}_1}(a_2, b_2, f(b_2), c_2, \mathbf{apd}_f(c_2)) \\ \mathbf{apd}^2_f(\mathrm{c}_2(a_3, b_3, c_3, d_3)) &= \tilde{\mathrm{c}_2}(a_3, b_3, f(b_3), c_3, \mathbf{apd}_f(c_3), d_3, \mathbf{apd}^2_f(d_3)) \end{aligned}$$

We have already shown the assumptions on $\tilde{\mathrm{c}_0}$ and $\tilde{\mathrm{c}_1}$. We also have

$$\begin{aligned} \tilde{\mathrm{c}_2} \quad : \quad & (a_3 : A_2) \to (b_3 : \mathrm{H}) \to (\tilde{b}_3 : C(b_3)) \to (c_3 : p_3 =_{\mathrm{H}} q_3) \\ & \to (\tilde{c}_3 : \mathrm{T}_0(p_3) =^{C}_{c_3} \mathrm{T}_0(q_3)) \to (d_3 : g_1 =_{p_4 =_{\mathrm{H}} q_4} h_1) \\ & \to \mathrm{T}_1(g_1) =^{\mathrm{T}_0(p_4) =^{\mathrm{H}}_{-} \mathrm{T}_0(q_4)}_{d_3} \mathrm{T}_1(h_1) \\ & \to \mathrm{T}_1(g_2) =^{\mathrm{T}_0(p_5) =^{\mathrm{H}}_{-} \mathrm{T}_0(q_5)}_{\mathrm{c}_2(a_3, b_3, c_3, d_3)} \mathrm{T}_1(h_2) \end{aligned}$$

Groupoid model of H

The interpretation of H is the groupoid $(\mathrm{H}_0, \mathrm{H}_1, \mathrm{H}_2)$, where

- H_0 is the inductively defined set of objects (elements, points).
- $\mathrm{H}_1(x, y)$ is the inductively defined family of set of arrows (identity proofs, paths)
- $\mathrm{H}_2(x, y, f, g)$ is the inductively defined family of set of 2-cells (identity proofs of arrows, surfaces, homotopies)

The objects of H

H_0 is inductively generated by a constructor for the object part of the point constructor

$$\mathrm{c}_{00} \quad : \quad (A_0)_0 \to \mathrm{H}_0 \to \mathrm{H}_0$$

The arrows of H

H_1 is inductively generated by:

- a constructor for the object part of the path constructor

$$\begin{aligned} \mathrm{c}_{10} \quad : \quad & (x \in (A_1)_0) \to (y \in \mathrm{H}_0) \\ & \to \mathrm{H}_1(p_0(x, y), q_0(x, y)) \to \mathrm{H}_1(p'_0(x, y), q'_0(x, y)) \end{aligned}$$

- a constructor for the arrow part of the point constructor:

$$\begin{aligned} \mathrm{c}_{01} \quad : \quad & (x, x' \in (A_0)_0) \to (A_0)_1(x, x') \to (y, y' \in \mathrm{H}_0) \\ & \to \mathrm{H}_1(y, y') \to \mathrm{H}_1(\mathrm{c}_{00}(x, y), \mathrm{c}_{00}(x', y')) \end{aligned}$$

- constructors for composition, identity, and inverse of paths

$$\begin{aligned} \circ \quad &: \quad (x, y, z \in \mathrm{H}_0) \to \mathrm{H}_1(x, y) \to \mathrm{H}_1(y, z) \to \mathrm{H}_1(x, z) \\ \mathrm{id} \quad &: \quad (x \in \mathrm{H}_0) \to \mathrm{H}_1(x, x) \\ (-)^{-1} \quad &: \quad (x, y \in \mathrm{H}_0) \to \mathrm{H}_1(x, y) \to \mathrm{H}_1(y, x) \end{aligned}$$

The surfaces of H

H_2 (representing equality of paths) is inductively generated by

- c_{20} – the object part of the surface constructor
- c_{11} – the arrow part of the path constructor
- c_{02} – the surface (preservation of equality of arrows) part of the point constructor:
- $\mathrm{c}_0^{\mathrm{id}}, \mathrm{c}_0^{\circ}$ – witnesses for the functor laws for the point constructor
- $\mathrm{tran}, \mathrm{refl}, \mathrm{sym}$ – witnesses that H_2 is a family of equivalence relations
- $\mathrm{w}_0, \mathrm{w}_1$ – witnesses that composition preserves equality
- $\alpha, \lambda, \rho, \iota_0, \iota_1$ – witnesses for the groupoid laws

Interpretation of formation rule

It's clear that $(\mathrm{H}_0, \mathrm{H}_1, \mathrm{H}_2)$ is a groupoid.

Interpretation of introduction rules

- The point constructor $\mathrm{c}_0 : A_0 \to \mathrm{H} \to \mathrm{H}$ is interpreted by the functor on groupoids with object part c_{00}, arrow part c_{01} and preservation of equality part c_{02}. The functor laws are witnessed by the constructors $\mathrm{c}_0^{\mathrm{id}}$ and c_0°.
- A groupoid interpreting $x =_{\mathrm{H}} y$ is a setoid and hence functors on such groupoids degenerate to setoid-maps. Hence, the path constructor

$$
\begin{aligned}
\mathrm{c}_1 \quad : \quad & (x : A_1) \to (y : \mathrm{H}) \to p(x, y) =_{\mathrm{H}} q(x, y) \\
& \to p_1(x, y) =_{\mathrm{H}} q_1(x, y)
\end{aligned}
$$

  is interpreted by the setoid map with underlying function c_{10} and preservation of equality part c_{11}.
- A groupoid interpreting $f =_{x =_{\mathrm{H}} x'} f'$ has only one object and one arrow (up to equality). Hence it suffices that the constructor c_2 is interpreted by c_{20}.

Interpretation of elimination and equality rules

We want to show that there exists a "dependent groupoid functor"

$$f : (x : \mathrm{H}) \to C(x)$$

such that

$$\begin{aligned}
f(\mathrm{c}_0(x, y)) &= \tilde{\mathrm{c}_0}(x, y, f(y)) \\
\mathbf{apd}_f(\mathrm{c}_1(x, y, z)) &= \tilde{\mathrm{c}_1}(x, y, f(y), z, \mathbf{apd}_f(z)) \\
\mathbf{apd}^2_f(\mathrm{c}_2(x, y, z, w)) &= \tilde{\mathrm{c}_2}(x, y, f(y), z, \mathbf{apd}_f(z), w, \mathbf{apd}^2_f(w))
\end{aligned}$$

Object and arrow part of f

- Object part $f_0 : (x \in \mathrm{H}_0) \to C_0(x)$ by

$$f_0(\mathrm{c}_{00}(x, y)) \quad = \quad (\tilde{c_0})_0(x, y, f_0(y)))$$

- Arrow part

$$f_1 \quad : \quad (x, x' \in \mathrm{H}_0) \to (g \in \mathrm{H}_1(x, x')) \to C_1'(g, f_0(x), f_0(x'))$$

where C_1' is a heterogenous version of arrow (between elements of different fibers). This is done by H_1-elimination:

$$f_1(\mathrm{c}_{10}(x, y, z)) \quad = \quad (\tilde{c_1})_0(x, y, f_0(y), z, f_1(p, q, z))$$

$$f_1(\mathrm{c}_{01}(x, x', e, y, y', d)) \quad = \quad (\tilde{c_0})_1(x, x', e, y, y', d, f_0(y), f_0(y'), f_1(y,$$

and clauses which say that f_1 maps an identity on H to an identity, a composition to a composition, and an inverse to an inverse.

Preservation of equality of arrows part of f

We define the 2-cell part

$$
\begin{aligned}
f_2 \quad : \quad & (x, x' \in \mathrm{H}_0) \to (g, g' \in \mathrm{H}_1(x, x')) \to (* \in \mathrm{H}_2(x, x', g, g')) \\
& \to C_2'(*, f_1(x, x', g), f_1(x, x', g'))
\end{aligned}
$$

where C_2' is a heterogenous notion of equality between elements in different fibres. This is proved by H_2-elimination.

∞-Hits?

Can the schemata for 1- and 2-hits be extended to arbitrary n-hits and also to ∞-hits?

- Can cubical type theory be extended with schema for hits with constructors of arbitrary dimensionality?
- Can these hits be interpreted in the Kan cubical set model?

A step on the way:

- Formulate 1- and 2-hits using face maps and degeneracies.
- Formulate setoids and groupoids as truncated Kan cubical sets.