The definitional symmetric cubical structure of types in type theory with equality defined by abstraction over an interval

Hugo Herbelin
INRIA - IRIF - University Paris Diderot

Abstract

Cohen, Coquand, Huber and Mörtberg [CCHM18] introduced a type theory whose equality type is defined as a (dependent) product over a formal notion of interval. This approach directly endows the tower of nested equalities over a type with a symmetric cubical structure whose equations over the operations of the structure hold definitionally.

We study a few properties of this structure from a typed perspective.

We consider a type theory with a universe \( U \) and a heterogeneous equality defined by dependent product over an interval (\cite{CCHM18}, Section 9). We start with an interval with no particular structure, besides supporting variables \( i, j, k, \ldots \) and formal endpoints 0 and 1 (i.e. interval expressions are defined by \( \tau ::= i \mid 0 \mid 1 \)). Typing contexts include declaration of interval variables. The rules for equality, essentially taken from \cite{CCHM18}, are the following ones:

\[
\begin{align*}
\Gamma \vdash \xi : A =_\emptyset B & \quad \Gamma \vdash t : A & \quad \Gamma \vdash u : B & \quad \Gamma \vdash t =_\xi u & \quad \Gamma \vdash p : t =_\xi u \\
\Gamma \vdash t =_\xi u : U & \quad \Gamma \vdash p 0 \equiv t & \quad \Gamma \vdash p 1 \equiv u \\
\Gamma, i \vdash t : A & \quad \Gamma \vdash v : t =_\xi u & \quad FV(\tau) \in \Gamma \\
\Gamma \vdash \lambda i. t : t[0/i] =_{\lambda i. A} t[1/i] & \quad \Gamma \vdash v \tau : \xi \tau 
\end{align*}
\]

where we write \( \hat{t} \) to abbreviate \( \lambda i. t \) for \( i \) not occurring in \( t \) (if \( t \) is of type \( A \), \( \hat{t} \) is a proof of \( t =_A t \) where \( \hat{A} \), this time with \( A \) of type \( U \), is itself a proof of \( A =_\emptyset A \)).

The type \( A =_\emptyset B \) can be seen as the type of lines connecting the types \( A \) and \( B \). Let us call its inhabitants line types. For \( \xi \) a line type between \( A \) and \( B \) and for \( t \) of type \( A \) and \( u \) of type \( B \), the type \( t =_\xi u \) can be seen as the type of lines between \( t \) of type \( A \) and \( u \) of type \( B \). An inhabitant of such a type is called a line and we say that it has line type \( \xi \).

Let us then consider types \( A, B, C, \) and \( D \), as well as lines \( \xi, \zeta, \phi \) and \( \psi \) relating these types as in the square drawn on the left below:

\[
\begin{array}{ccc}
A & \phi & C \\
\xi \downarrow & \equiv & \zeta \\
B & \psi & D \\
\end{array}
\]

This square can be identified with the type \( \xi =_{\phi \equiv \psi} \zeta \) where \( \phi \equiv_{\psi} \zeta \) abbreviates \( \lambda j. (\phi j =_{\xi j} \psi j) \).

Its inhabitants we call square types.

Let us next consider \( t, u, v, \) and \( w \) of type \( A, B, C, \) and \( D \) respectively, and \( p, q, r \) and \( s \) lines between these points as drawn in the square above on the right, and \( E \) a square type, i.e. a proof of type \( \xi =_{\phi \equiv \psi} \zeta \) for some \( \xi, \zeta, \phi \) and \( \psi \) connecting \( A, B, C \) and \( D \) as in the figure.

One can consider the type \( P =_{r \equiv_E s} q \), where \( r \equiv_E s \) again abbreviates \( \lambda j. (r j =_{E j} s j) \). This can be seen as the type of squares with edges \( p, q, r \) and \( s \) and square type \( E \).
We use the abbreviation \( t \equiv^n_\xi u \triangleq \lambda i_1 \ldots i_n. (ti_1 \ldots i_n = \xi i_1 \ldots i_n u i_1 \ldots i_n) \) and define 3-dimensional cube types as inhabitant of types of the form \( E = F^G \) on their boundaries, the previous nesting process allows to define a type of \( \Psi \) of the language. Examples of operations include:

- \( \epsilon(t) \triangleq \tilde{t} \), of dimension 0 to 1 with both \( \Psi_0(t) \) and \( \Psi_1(t) \) returning the empty list of faces;
- degeneracies/reflexivity: \( \alpha = \beta \), of dimension 2 to 2, with \( \Psi_0(\alpha) \triangleq [\lambda i. \alpha i, \lambda \alpha ii] \) and \( \Psi_1(\alpha) \triangleq [\lambda i. \alpha ii, \lambda i i ii] \);
- transpositions/interchange: \( \sigma(\alpha) \triangleq \lambda i j. \alpha ji \), of dimension 2 to 2, with \( \Psi_0(\alpha) \triangleq [\lambda i. \alpha i, \lambda i i \alpha ii] \) and \( \Psi_1(\alpha) \triangleq [\lambda i i \alpha ii, \lambda i. \alpha ii ii] \);
- left (resp. right) connections: \( \Gamma^+(p) \) (resp. \( \Gamma^-(p) \)) which can be taken as axioms, of dimension 1 to 2, with both \( \Psi_0(p) \) and \( \Psi_1(p) \) being \([p, p^1] \) (resp. \([\tilde{p}, \tilde{p^1}] \));
- reversals/inverses: \( p^{-1} \) which can be taken as an axiom, of dimension 1 to 1 with \( \Psi_0(p) \triangleq p1 \) and \( \Psi_1(p) \triangleq p0 \);
- diagonals: \( \Delta(\alpha) \triangleq \lambda i. \alpha ii \), of dimension 2 to 1 with \( \Psi_0(p) \triangleq \alpha 00 \) and \( \Psi_1(p) \triangleq \alpha 11 \).

Operations from dimension 1 to some dimension \( p \) can be internalized as algebraic operations of arity \( p \) on the interval. For instance, reversal and connections can be obtained, as in [CCHM18], by extending the interval with \( \tau := \ldots | - \tau | - \tau \wedge \tau | - \tau \vee \tau \) and defining \( p^{-1} \triangleq \lambda i. p(-i) \), as well as \( \Gamma^+(p) \triangleq \lambda i j. p(i \wedge j) \) and \( \Gamma^-(p) \triangleq \lambda i j. p(i \vee j) \). Using iterated congruence, as defined by:

\[
\tilde{\Phi}^m(t) \triangleq \Phi(t) \quad \tilde{\Phi}^{m+1}(t) \triangleq \lambda i. \tilde{\Phi}^m(ti) \quad i \text{ taken fresh}
\]

any operation \( \Phi \) from dimension \( n \) can be extended into an operation \( \Phi_m \) acting on cubes of dimension at least \( m+n \). For instance, for \( \aleph \) of dimension \( q \geq 1 \) and \( 0 \leq m < q \), \( \partial^+_{m}(:=) \triangleq \tilde{\partial}^+_m(:=) \) is the \( m \)-th left face operation of the cubical structure.

Note in passing that any \( n \)-cube of type \([\alpha_1, \ldots, \alpha_n] \approx \xi [\beta_1, \ldots, \beta_n] \) can alternatively be seen as a \( p \)-cube of type \([\alpha_{p+1}, \ldots, \alpha_n] \approx \xi [\beta_{p+1}, \ldots, \beta_n] \) \([\beta_1, \ldots, \beta_p] \) for \( 0 \leq p \leq n \). Hence, any operation acting on a \( p \)-cube directly acts also on an \((p+q)\)-cube.

Operations can also be extended to take several arguments, with composition or tensor product as examples.

References