A very short note on homotopy λ-calculus

Vladimir Voevodsky

September 27, 2006

The homotopy λ-calculus is a hypothetical (at the moment) type system. To some extent one may say that $H\lambda$ is an attempt to bridge the gap between the "classical" type systems such as the ones of PVS or HOL Light and polymorphic type systems such as the one of *Coq*. The main problem with the polymorphic type systems lies in the properties of the equality types. As soon as we have a universe U of which $Prop$ is a member we are in trouble. In the Boolean case, $Prop$ has an automorphism of order 2 (the negation) and it is clear that this automorphism should correspond to a member of $Eq(U; Prop, Prop)$. However, as far as I understand there is no way to produce such a member in, say, *Coq*. A related problem looks as follows. Suppose $T, T' : U$ are two type expressions and there exists an isomorphism $T \to T'$ (the later notion of course requires the notion of equality for members of T and T'). Clearly, any proposition which is true for T should be true for T' i.e. for all functions $P : U \to Prop$ one should have $P(T) = P(T')$. Again as far as I understand this can not be proved in *Coq* no matter what notion of equality for members of T and T' we use.

Here is the general picture as I understand it at the moment. Let us consider the type system TS which is generated by the sequents

$$\vdash U_i : U_{i+1}$$

(for $i = -1, 0, 1, \ldots$) and the rules:

1. $$\frac{\Gamma \vdash T : U_i}{\Gamma \vdash T : U_{i+1}} \qquad \frac{\Gamma \vdash T : U_i}{\Gamma \vdash T : Type}$$

2. The usual dependent $\prod$-rules (inside each U_n)

3. The usual dependent $\sum$-rules with strong elimination (inside each U_n)[1]

The system $H\lambda$ is supposed to be an extension of TS. In $H\lambda$, U_{-1} becomes the empty type $\emptyset$ and U_0 becomes $Prop$. The natural numbers are defined (see (1) below) in terms of U_1.

Let CC be the contexts category of TS. By a model of TS with values in a category D, I mean a functor $CC \to D$ which "preserves the relevant structures". The main observation is that there is a canonical model M of TS with values in the usual homotopy category H provided that we consider homotopy types based on a sufficiently large universe of sets. To define this model one starts with a not-so-canonical model N of TS with values in the category of spaces (actually simplicial sets, but I will speak of spaces since they provide a more familiar model for homotopy types) and then sets M to be the composition of N with the projection $Spc \to H$. The main properties of N are are follows.

[1] We may also consider systems TS_X where X is any "recursive" partially ordered set such that U_x is defined for any $x \in X$ and the rules are modified accordingly. If X is just a finite set with the trivial ordering then it seems that TS_X will be just the usual typed λ-calculus with products generated by n primitive types. The first system with real dependencies is TS_X where $X = \{0, 1\}$ with the usual ordering.

1. By definition N takes a context Γ to a space $N(\Gamma)$.
2. A sequent of the form $\Gamma \vdash T : Type$ (where T is an expression) defines a morphism

$$(\Gamma, x : T) \to \Gamma$$

   in CC. Morphisms of this type go to *fibrations*

$$N(\Gamma; T) : N(\Gamma, x : T) \to N(\Gamma),$$

3. A sequent of the form $\Gamma \vdash t : T$ (where T and t are expressions) defines a morphism

$$\Gamma \to (\Gamma, x : T)$$

   in CC. Morphisms of this type go to *sections*

$$N(\Gamma; T, t) : N(\Gamma) \to N(\Gamma, x : T)$$

   of $N(\Gamma; T)$.

Given $\Gamma \vdash P : Type$ and $\Gamma, x : P \vdash Q : Type$ we can form $\Gamma \vdash \prod x : P.Q$ and $\Gamma \vdash \sum x : P.Q$. On the model level our data defines two fibrations

$$N(\Gamma, x : P, y : Q) \xrightarrow{q} N(\Gamma, x : P) \xrightarrow{p} N(\Gamma)$$

The fibration

$$N(\Gamma, z : \prod x : P.Q) \to N(\Gamma)$$

is the "$p_*(q)$". Its fiber over $x \in N(\Gamma)$ is the space of sections (continuous ones!) of the fiber of q over x.

The fibration

$$N(\Gamma, z : \sum x : P.Q) \to N(\Gamma)$$

is the "$p_!(q)$". It is simply the composition of p and q. The meaning of term constructors associated with $\sum$ and $\prod$ is the obvious one. If we took a model with values in *Sets* where all maps are fibrations we would get the usual rules for interpretation of $\sum$ and $\prod$ but formulated in a slightly unusual way.

The rigorous description of the value of N on U_n's is complicated. Up to homotopy equivalence, the space $N(U_n)$ is the nerve of the n-groupoid of $(n-1)$-groupoids in the ZF with $n-2$ universes (see below for the explicit form in the case $n \leq 1$). Alternatively, one may say that U_n is the base of the universal fibration whose fibers are $(n-1)$-types which lie in ZF with $n-2$ universes (so that itself it lies in the ZF with $n-1$ universe. The equivalence of these two points of view follows from the fact that n-groupoids are the same as n-homotopy types.

In particular,

$$M(U_{-1}) = \emptyset$$

and

$$M(U_0) = \{0, 1\}.$$

We further have

$$M(U_1) = \coprod_{n\geq 0} BS_n$$

where BS_n is the classifying space of the permutation group on n elements (BS_0 is empty, BS_1 is one point and BS_2 is homotopy equivalent to $\mathbf{R}P^\infty = B\mathbf{Z}/2$). In particular, $\pi_0(M(U_1)) = \mathbf{N}$ and one uses U_1 to define the type of natural numbers in $H\lambda$. As far as I understand at the moment $M(U_2)$ is $\coprod_{X\in u_2} BAut(X)$ where u_2 is the set of equivalence classes of all groupoids with sets of morphisms and objects being ZF-sets. Starting with U_2 one needs ZF with universes in order for the model to be defined. The model of U_3 is the nerve of the 3-groupoid of 2-groupoids in ZF with one universe.

This model is very "incomplete" in the sense that there are many type expressions T such that $M(T)$ is non-empty while T has no terms in TS. This is of course unavoidable because of the Goedel's theorem. However, some of these incompletenesses are of a special kind. For example $M(U_{-1}) = \emptyset$ hence we may add the empty type rule

$$\frac{\Gamma \vdash c : U_{-1} \quad \Gamma \vdash T : Type}{\Gamma \vdash \iota(c, T) : T}$$

which expresses the fact that if the empty type is inhabited in a context then any other type is. It does not look provable in TS.

Other examples of such rules involve the equality types. Given a valid type expression $T : U_n$ and two term expressions $t1, t2 : T$ we get on the level of models a space $X = M(T)$ (up to homotopy) and two points $x1, x2 \in X$. One of the most important observations concerning the picture outlined so far is that it is possible to define equality (equivalence) types $Eq(T; t1, t2)$ in TS such that the model of $Eq(T; t1, t2)$ is (homotopy equivalent to) the space $P(X; x1, x2)$ of paths from $x1$ to $x2$ in X.

The definition proceeds in the following steps:

Define the contractibility on the level of TS. Set

$$true = (U_{-1} \to U_{-1}) : U_0$$

$$false = U_{-1} : U_0.$$

For $T, T' : U_0$ set

$$Equiv(T, T') = (T \to T') \times (T' \to T).$$

For $T : U_n$ set

$$Contr(T) = \prod F : U_n \to U_0.Equiv(F(T), F(true))$$

then $M(Contr(T)) \neq \emptyset$ iff $M(T)$ belongs to the same connected component of $M(U_n)$ as $M(true)$ i.e. if $M(T)$ is a contractible space. In that case $M(Contr(T))$ is itself contractible.

Define representable functors on the level of TS. Suppose $T : U_n$ is a type (expression). I want to think of its model $X = M(T)$ as of the nerve of some n-groupoid in U_n. The members of T correspond to objects. For $T = U_n$ we get the groupoid of all groupoids. Functions $T \to U_n$ correspond to functors from T to the groupoid of all groupoids. Among these functors there are representable ones i.e. we have the homotopy type $Rep(T)$ which maps to $T \to U_n$. For $F : T \to U_n$ set

$$rep(F) = Contr(\sum t : T.F(T)).$$

One verifies that on the level of models $rep(F) \neq \emptyset$ iff F is representable. Set

$$Rep(T) = \sum F : T \to U_n.rep(F).$$

then the model of $Rep(T)$ is the space of representable functors on T. By abuse of notation I will write $F(t) : U_n$ instead of the formal $(\pi F)(t)$ for $F : Rep(T)$ and $t : T$.

Define the equality types. For $T : U_n$ and $T1, t2 : T$ one sets:

$$Eq(T; t1, t2) = \prod F : Rep(T).F(t1) \to F(t2)$$

where I write $F(t)$ for $F : Rep(T)$ instead of the correct but long $(\pi F)(t)$.

Theorem 1 *There is a homotopy equivalence*

$$M(Eq(T; t1, t2)) = P(M(T); M(t1), M(t2)).$$

Once the equality types (path spaces) are defined many other constructions familiar on the model level can be formulated on the level of the type system. The first thing to define is the *level* "filtration" on type expressions or, equivalently on the types U_n. The model of U_n has a natural filtration by subspaces $U_{n,k}$, $k = 0, \dots, n$ where $U_{n,k}$ is (the nerve of) the k-groupoid of $(k-1)$-groupoids in the universe U_n. In particular $U_{n,1}$ is the (nerve of) the usual groupoid of sets in U_n and their isomorphisms. We define a (-1)-groupoid as a set where any two elements are equal i.e. one of the two sets $\emptyset$ and pt. Hence for any $n \geq 0$ the model of $U_{n,0}$ is the two point set $\{0, 1\} = \{true, false\}$.

$$\begin{tikzcd}
U_{0,0} \arrow[r, equal] & U_{1,0} \arrow[r, equal] \arrow[d] & U_{2,0} \arrow[r, equal] \arrow[d] & U_{3,0} \arrow[r, equal] \arrow[d] & \dots \\
 & U_{1,1} \arrow[r] & U_{2,1} \arrow[r] \arrow[d] & U_{3,1} \arrow[r] \arrow[d] & \dots \\
 & & U_{2,2} \arrow[r] & U_{3,2} \arrow[r] \arrow[d] & \dots \\
 & & & U_{3,3} \arrow[r] & \dots
\end{tikzcd}$$

All the arrows are inclusions with the image being a disjoint union of some of the connected components of the target and the usual arguments a-la Russell's paradox imply that except for the ones marked as equalities the arrows are proper inclusions e.g. $U_{2,1}$ (which is responsible for sets in U_2) is strictly larger than $U_{1,1}$ (which is responsible for sets in U_1) etc.

To get $U_{n,k}$'s as type expressions we first define type expressions $Lv_k(T) : U_0$ for $T : U_n$ which are "indicator functions" for $U_{n,k}$ setting:

$$Lv_{-1}(T) = Contr(T)$$

and for $k \geq 0$

$$Lv_k(T) = \prod t1 : T. \prod t2 : T.Lv_{k-1}(Eq(T; t1, t2)).$$

Then

$$U_{n,k} = \sum T : U_n.Lv_k(T).$$

One verifies easily that this definition is consistent with the model level definition given above. We will also use the following notations. For $F : T' \to T$ and $t : T$ set

$$Hfiber(F, t) = \sum t' : T'.Eq(T; t, F(t'))$$

one verifies easily that the model of $Hfiber$ is the homotopy fiber of F over t. Set further

$$isheq(F) = \prod t : T.Contr(hfiber(F, t)).$$

This is a truth value and the model of $isheq(f)$ is $true$ iff the model of f is a homotopy equivalence. Set further

$$Heq(T', T) = \sum F : T' \to T.isheq(F)$$

then $M(Heq(T', T))$ is the space of homotopy equivalences from T' to T. For a type expression T set:

$$\Pi_{-1}(T) = (T \to U_{-1}) \to U_{-1}$$

it is a truth value and on the model level $\Pi_{-1}(T) = true$ iff T is not empty. For $F : T' \to T$ set

$$Im(F) = \sum t : T.\Pi_{-1}(Hfiber(F, t)).$$

The model of $Im(F)$ is the union of connected components of T whose pre-image under F is non-empty. Set

$$ev(T', T) = \lambda t : T.\lambda F : T' \to T.F(t) : T' \to ((T' \to T) \to T)$$

$$\Pi_0(T) = Im(ev(T, U_0)).$$

The model of $\Pi_0(T)$ is the set of connected components of T.

We can now give more examples of things which hold on the model level but (probably) can not be proved on the level of TS or even TS with the empty type rule.

1. The natural maps $U_{n,0} \to U_{n+1,0}$ are equivalences on the model level for $n \geq 0$. It seems to be unprovable in TS. To fix it one may add the rule

$$\frac{\Gamma \vdash T : Type \quad \Gamma \vdash a : Lv_0(T)}{\Gamma \vdash T : U_0}.$$

   Alternatively one can impose stabilization together with the Boolean rule by adding a term constructor

$$\frac{\Gamma \vdash T : Type \quad \Gamma \vdash a : Lv_0(T)}{\Gamma \vdash boo(T, a) : ((T \to \emptyset) \to \emptyset) \to T}$$

2. Set
$$\mathbf{N} = \Pi_0(U_1). \tag{1}$$
The model of $\mathbf{N}$ is of course $\mathbf{N}$ – the set of natural numbers. It is not clear however to what extend $\mathbf{N}$ is a natural numbers object in the sense of type theory.

3. For $T : U_n$ we may consider T also as a member of U_{n+1}. Thus we have two definitions of $Rep(T)$ one using U_n and another one using U_{n+1}. They should agree i.e. any $F : T \to U_{n+1}$ such that $rep(F) = true$ should factor through U_n. More precisely there are two expressions $Rep_n(T)$ and $Rep_{n+1}(T)$ and it should be possible to construct a function $Rep_{n+1}(F) \to Rep_n(F)$ which is "inverse" to the obvious one going in the opposite direction.

4. Given $T, T' : U_n$ there are two different expressions which both model to the space of equivalences from T' to T. One is $Eq(U_n; T', T)$ and another one is $Heq(T', T)$. So we should have an equivalence
$$Heq(T', T) \to Eq(U_n; T', T).$$
Again it is unclear how to construct it on the level of the type system.

A subtle thing about imposing all these properties on TS is that while they all hold for M it is not clear which ones one may get on the level of N. In particular the stabilization of $U_{0,k}$'s does not hold for the version of N which I have been considering. For example $N(U_{0,2})$ is much smaller as a space then $N(U_{0,3})$ since there are many more one point sets in ZF with a universe then there are in "pure" ZF.

The context category $CH\lambda$ of $H\lambda$ has a structure reminiscent of a Quillen model structure or rather of the structure of a category with fibrations and weak equivalences considered by Baues in "Algebraic Homotopy". The associated homotopy category HH is some sort of a free homotopy category.

Originally, I was considering a different approach to $H\lambda$ where the equality types where introduced as "primitives" along with $\sum$ and $\prod$ and the universes where "defined" but it seems to me now that it is nicer to start with $\sum$, $\prod$ and universes and define the equality types later. What is left from this earlier stage is certain understanding of which properties of/structures on the equality types might be sufficient to ensure that they behave nicely (e.g. that for any $t : T$, $\pi_1(T; t) = \Pi_0(Eq(T; t, t))$ is a group or that there is a long exact sequence of "homotopy groups" associated to a fibration).

The advantage of $H\lambda$ and its homotopy-theoretic model over the less sophisticated type systems is that it better reflects the way mathematicians envision "types" corresponding to mathematical structures of higher level. For example if we fix the size of the universe n and write in the usual way the type expression for, say, the type $Gr(U_n)$ of groups in U_n then the model of $Gr(U_n)$ will be (the nerve of) the groupoid of groups in the universe U_n and their isomorphisms. Similarly, if we write down the definition of a category in a proper way then the model of $Cat(U_n)$ will be (the nerve of) the 2-groupoid of categories in U_n, their equivalences and natural isomorphisms between equivalences. Moreover, any construction on categories described in the language of $H\lambda$ is automatically "invariant" under equivalences of categories. E.g. any function we can describe in $H\lambda$ from $Cat(U_n)$ to $Gr(U_n)$ will on the model level correspond to a construction which produces a group from a category which maps equivalences between categories to isomorphisms between the corresponding groups. In the usual type systems we can do something like that for types of "level 1" i.e. sets with structures but not for higher levels (e.g. categories).

At the moment much of what I said above is at the level of conjectures. Even the definition of the model of TS in the homotopy category is non-trivial. Similarly, the definition of equality types in terms of universes is rather involved and I am not sure which of the properties of these types have to be imposed so that the rest will follow.